

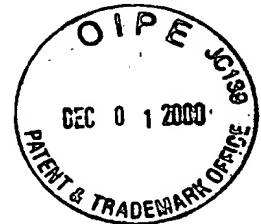
09/650.999

(translation of the front page of the priority document of
Japanese Patent Application No. 2000-258914)

PATENT OFFICE
JAPANESE GOVERNMENT

RECEIVED
DEC 06 2000
Technology Center 2600

This is to certify that the annexed is a true copy of the
following application as filed with this Office.



Date of Application: August 29, 2000

Application Number : Patent Application 2000-258914

Applicant(s) : Canon Kabushiki Kaisha

September 29, 2000

Commissioner,
Patent Office

Kouzo OIKAWA

Certification Number 2000-3080009

CERTIFIED COPY OF
PRIORITY DOCUMENT

CFM1990US

09/650.999

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出 願 年 月 日

Date of Application:

2000年 8月29日

出 願 番 号

Application Number:

特願2000-258914

出 願 人

Applicant(s):

キヤノン株式会社

RECEIVED

DEC 06 2000

Technology Center 2600

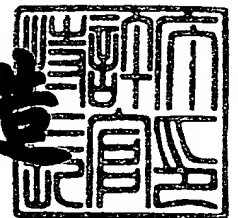


BEST AVAILABLE COPY

2000年 9月29日

特許庁長官
Commissioner,
Patent Office

及 川 耕 造



出証番号 出証特2000-3080009

【書類名】 特許願

【整理番号】 4010027

【提出日】 平成12年 8月29日

【あて先】 特許庁長官殿

【国際特許分類】 G06K 15/00

【発明の名称】 画像処理システム及びその制御方法

【請求項の数】 21

【発明者】

 【住所又は居所】 東京都大田区下丸子3丁目30番2号 キヤノン株式会社
 社内

 【氏名】 種田 淳

【特許出願人】

 【識別番号】 000001007

 【氏名又は名称】 キヤノン株式会社

【代理人】

 【識別番号】 100076428

 【弁理士】

 【氏名又は名称】 大塚 康德

 【電話番号】 03-5276-3241

【選任した代理人】

 【識別番号】 100101306

 【弁理士】

 【氏名又は名称】 丸山 幸雄

 【電話番号】 03-5276-3241

【選任した代理人】

 【識別番号】 100115071

 【弁理士】

 【氏名又は名称】 大塚 康弘

 【電話番号】 03-5276-3241

【手数料の表示】

【予納台帳番号】 003458

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0001010

【プールの可否】 要

【書類名】 明細書

【発明の名称】 画像処理システム及びその制御方法

【特許請求の範囲】

【請求項 1】 画像入力装置と画像出力装置とをシリアルバスにより接続した画像処理システムであって、

前記画像入力装置は、

第 1 の形式の画像データを前記画像出力装置へ送信する第 1 の通信手段と、

送信する画像データの形式を変換すべきか否かを判定する判定手段と、

該判定結果に基づいて、前記第 1 の形式の画像データを第 2 の形式に変換する第 1 の変換手段と、

を有することを特徴とする画像処理システム。

【請求項 2】 前記画像出力装置は、

前記画像入力装置から転送されてきた画像データを受信する第 2 の通信手段と

、
該受信した画像データを所定容量のバッファに一時保持する保持手段と、

前記バッファに保持された画像データが前記第 1 の形式であれば前記第 2 の形式に変換する第 2 の変換手段と、

前記第 2 の形式の画像データを順次出力する出力手段と、

を有することを特徴とする請求項 1 記載の画像処理システム。

【請求項 3】 前記第 1 の形式は圧縮されたデータ形式であり、前記第 2 の形式は該第 1 の形式の画像データを伸長したデータ形式であることを特徴とする請求項 2 記載の画像処理システム。

【請求項 4】 前記第 1 の形式は J P E G 形式であることを特徴とする請求項 3 記載の画像処理システム。

【請求項 5】 前記画像入力装置における判定手段は、前記画像出力装置における前記バッファの空き状況に基づいて、画像データの形式を変換すべきか否かを判定することを特徴とする請求項 2 乃至 4 のいずれかに記載の画像処理システム。

【請求項 6】 前記判定手段は、前記バッファに空きがない場合に、画像デ

ータの形式を変換すべきと判定することを特徴とする請求項 5 記載の画像処理システム。

【請求項 7】 前記判定手段は、前記転送手段において前記シリアルバスにおけるビジー状態が検出された場合に、画像データの形式を変換すべきと判断することを特徴とする請求項 6 記載の画像処理システム。

【請求項 8】 前記第 2 の通信手段は、前記バッファの空き状況を示すバッファ情報を前記画像入力装置へ通知し、

前記判定手段は、該バッファ情報に基づいて、画像データの形式を変換すべきか否かを判定することを特徴とする請求項 6 記載の画像処理システム。

【請求項 9】 前記第 2 の通信手段は、前記バッファの空き状況に基づいて前記画像入力装置へ画像データの形式変換要求を発行し、

前記判定手段は、該形式変換要求をうけた場合に、画像データの形式を変換すべきと判断することを特徴とする請求項 1 乃至 4 のいずれかに記載の画像処理システム。

【請求項 10】 前記第 2 の通信手段は、前記バッファに空きがない場合に、前記形式変換要求を発行することを特徴とする請求項 9 記載の画像処理システム。

【請求項 11】 前記判定手段は、前記画像データの形式を変換すべきか否かを所定ブロック単位で判定し、

前記変換手段は、前記判定手段によって変換すべきと判定されたブロック以降の全ブロックについて、第 2 の形式に変換することを特徴とする請求項 1 乃至 10 のいずれかに記載の画像処理システム。

【請求項 12】 前記判定手段は、前記画像データの形式を変換すべきか否かを所定ブロック単位で判定し、

前記変換手段は、前記判定手段によって変換すべきと判定されたブロックのみについて、第 2 の形式に変換することを特徴とする請求項 1 乃至 10 のいずれかに記載の画像処理システム。

【請求項 13】 前記第 1 の通信手段は、前記画像データにおける第 1 番目のブロックについては、前記第 1 の形式で送信することを特徴とする請求項 1 1

又は 1 2 記載の画像処理システム。

【請求項 1 4】 前記画像入力装置は更に、

前記画像データにおける第 1 番目のブロックについて、前記第 1 の変換手段と前記第 2 の変換手段における処理能力を比較し、該処理能力の高い方で変換処理を行うことを決定する決定手段を有することを特徴とする請求項 1 1 又は 1 2 記載の画像処理システム。

【請求項 1 5】 前記シリアルバスは、IEEE1394規格に適合または準拠するバスであることを特徴とする請求項 1 記載の画像処理システム。

【請求項 1 6】 前記シリアルバスは、USB規格に適合または準拠するバスであることを特徴とする請求項 1 記載の画像処理システム。

【請求項 1 7】 画像入力装置と画像出力装置とをシリアルバスにより接続した画像処理システムの制御方法であって、

前記画像入力装置において、

第 1 の形式の画像データを前記画像出力装置へ送信する第 1 の通信工程と、

送信する画像データの形式を変換すべきか否かを判定する判定工程と、

該判定結果に基づいて、前記第 1 の形式の画像データを第 2 の形式に変換する第 1 の変換工程と、

を有することを特徴とする画像処理システムの制御方法。

【請求項 1 8】 前記画像出力装置において、

前記画像入力装置から転送されてきた画像データを受信する第 2 の通信工程と

該受信した画像データを所定容量のバッファに一時保持する保持工程と、

前記バッファに保持された画像データが前記第 1 の形式であれば前記第 2 の形式に変換する第 2 の変換工程と、

前記第 2 の形式の画像データを順次出力する出力工程と、

を有することを特徴とする請求項 1 7 記載の画像処理システムの制御方法。

【請求項 1 9】 前記第 1 の形式は圧縮されたデータ形式であり、前記第 2 の形式は該第 1 の形式の画像データを伸長したデータ形式であることを特徴とする請求項 1 8 記載の画像処理システムの制御方法。

【請求項 2 0】 前記判定工程においては、前記画像出力装置における前記バッファの空き状況に基づいて、画像データの形式を変換すべきか否かを判定することを特徴とする請求項 1 8 記載の画像処理システムの制御方法。

【請求項 2 1】 画像入力装置と画像出力装置とをシリアルバスにより接続した画像処理システムにおける制御プログラムを記録した記録媒体であって、該制御プログラムは少なくとも、

前記画像入力装置において、

第 1 の形式の画像データを前記画像出力装置へ送信する第 1 の通信工程のコードと、

送信する画像データの形式を変換すべきか否かを判定する判定工程のコードと

、
該判定結果に基づいて、前記第 1 の形式の画像データを第 2 の形式に変換する第 1 の変換工程のコードと、

を有することを特徴とする記録媒体。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は画像処理システム及びその制御方法に関し、特に画像入力装置と画像形成装置とを接続した画像処理システム及びその制御方法に関する。

【0 0 0 2】

【従来の技術】

近年のデジタル画像処理装置における性能の向上及び低価格化に伴い、デジタルカメラ等の画像入力装置とプリンタ等の画像形成装置とを接続した画像処理システムが普及している。このような画像処理システムにおいて画像データを処理する際には、画像入力装置において入力した画像データを圧縮し、該圧縮データを所定のインタフェースを介して画像形成装置へ転送する。そして、画像形成装置においては受信した圧縮データに対して伸長その他の画像処理を施した後に、印刷出力を行っていた。

【0 0 0 3】

【発明が解決しようとする課題】

しかしながら、上記従来の画像処理システムにおいては、画像入力から印刷終了までに要するトータルの処理時間（以降、スループットと称する）は、主にデータの転送時間と、画像形成装置側における圧縮データの伸長等の画像処理時間、及び印刷処理時間に依存しており、画像入力装置側における処理はトータルスループットにはほとんど影響しない。

【0004】

これは即ち、画像形成装置が動作している間に、画像入力装置では動作が停止しているという現象が発生することを意味し、システムを構成する機器間におけるパフォーマンスのバランスが悪いという問題あった。

【0005】

本発明は上記問題点を解決するためになされたものであり、画像入力装置と画像形成装置を接続した画像処理システムにおいて、転送データの形式を適切に変更することによって、トータルスループットを向上させる画像処理システム及びその制御方法を提供することを目的とする。

【0006】

【課題を解決するための手段】

上記目的を達成するための一手段として、本発明の画像処理システムは以下の構成を備える。

【0007】

即ち、画像入力装置と画像出力装置とをシリアルバスにより接続した画像処理システムであって、前記画像入力装置は、第1の形式の画像データを前記画像出力装置へ送信する第1の通信手段と、送信する画像データの形式を変換すべきか否かを判定する判定手段と、該判定結果に基づいて、前記第1の形式の画像データを第2の形式に変換する第1の変換手段と、を有することを特徴とする。

【0008】

また、前記画像出力装置は、前記画像入力装置から転送されてきた画像データを受信する第2の通信手段と、該受信した画像データを所定容量のバッファに一時保持する保持手段と、前記バッファに保持された画像データが前記第1の形式

であれば前記第 2 の形式に変換する第 2 の変換手段と、前記第 2 の形式の画像データを順次出力する出力手段と、を有することを特徴とする。

【 0 0 0 9 】

例えば、前記第 1 の形式は圧縮されたデータ形式であり、前記第 2 の形式は該第 1 の形式の画像データを伸長したデータ形式であることを特徴とする。

【 0 0 1 0 】

例えば、前記画像入力装置における判定手段は、前記画像出力装置における前記バッファの空き状況に基づいて、画像データの形式を変換すべきか否かを判定することを特徴とする。

【 0 0 1 1 】

【発明の実施の形態】

以下、本発明に係る一実施形態について、図面を参照して詳細に説明する。

【 0 0 1 2 】

<第 1 実施形態>

図1は、本発明を適用する画像処理システムの構成を示す図であり、1は画像データを入力するデジタルカメラ、2はデジタルカメラ1から入力された画像データを印刷出力するプリンタである。デジタルカメラ1及びプリンタ2は、それぞれのIEEE1394により規定されるインタフェース(I/F)部3、4により、ケーブル16を介して直結されている。

【 0 0 1 3 】

デジタルカメラ1は、画像入力処理を統括的に制御するCPU5、制御用プログラム等が格納されているROM6、一時記憶等を行うRAM7、実際に画像データの入力を行う画像入力部8、該入力された画像データに対して加工処理等を施す画像処理部9、及びI/F部3等により構成される。

【 0 0 1 4 】

一方、プリンタ2は、印刷処理を統括的に制御するCPU10、制御用プログラム等が格納されているROM11、一時記憶等を行うRAM12、印刷データに対して加工処理等を施すデータ処理部13、加工結果をプリンタ駆動部15へ引き渡す出力部14、実際の印刷処理を行うプリンタ駆動部15、及びI/F部4等により構成される。

【 0 0 1 5 】

上述したデジタルカメラ1及びプリンタ2におけるI/F部3, 4は互いに同様の構成からなり、図2にその詳細ブロック構成を示す。

【 0 0 1 6 】

同図において、201は物理層における動作であるデータ転送入出力、アービトレーション、及び送信データ符号のエンコード/デコード、等の機能を有するPHYチップ部（フィジカルレイアチップ）である。202はリンク層における動作である非同期データ転送、アイソクロナスデータ転送サポート、等の機能を有するLINKチップ部（リンクレイアチップ）である。203はI/F部全体を制御するCPU、204は制御用プログラム等が格納されているROM、205は一時記憶等を行うRAM、206はI/F部3, 4を、デジタルカメラ1, プリンタ2の内部バスと接続するI/F部である。

本実施形態においては、IEEE1394-1995規格（以下、「IEEE1394規格」と称する）により規定されるインタフェース（以下、「1394シリアルバス」と称する）を使用してデータ通信を行うことを前提とする。従って、予め1394シリアルバスの概要を以下に説明する。尚、IEEE1394規格についての詳細は、1996年の8月30日にIEEE(The Institute of Electrical and Electronics Engineers, Inc.)から出版された「IEEE Standard for a High Performance Serial Bus」に記述されている。

【 0 0 1 7 】

【IEEE1394の概要】

家庭用デジタルVTRやデジタルビデオディスク(DVD)の登場に伴い、ビデオデータやオーディオデータ（以下、まとめて「AVデータ」と呼ぶ）など、リアルタイムかつ情報量の多いデータを転送する必要があるが生じている。AVデータをリアルタイムに、PCへ転送したり、その他のデジタル機器に転送するには、高速のデータ転送能力をもつインタフェースが必要になる。そういった観点から開発されたインタフェースが1394シリアルバスである。

【 0 0 1 8 】

図3に1394シリアルバスを用いて構成されるネットワークシステムの例を示す。このシステムは機器AからHを備え、A-B間、A-C間、B-D間、D-E間、C-F間、C-G

間、およびC-H間がそれぞれ1394シリアルバス用のツイストペアケーブルで接続されている。これらの機器AからHの例としては、パソコンなどのホストコンピュータ装置、および、コンピュータ周辺機器である。コンピュータ周辺機器としては、デジタルVCR、DVDプレーヤ、デジタルスチルカメラ、ハードディスクや光ディスクなどのメディアを用いる記憶装置、CRTやLCDのモニタ、チューナ、イメージスキャナ、フィルムスキャナ、プリンタ、MODEM、ターミナルアダプタ(TA)などコンピュータ周辺機器のすべてが対象になる。なお、プリンタの記録方式は、レーザビームやLEDを用いた電子写真方式、インクジェット方式、インク溶融型や昇華型の熱転写方式、感熱記録方式など、どんな方式でも構わない。

【 0 0 1 9 】

各機器間の接続は、ディジーチェーン方式とノード分岐方式との混在が可能であり、自由度の高い接続を行うことができる。また、各機器はそれぞれIDを有し、互いにIDを認識し合うことによって、1394シリアルバスで接続された範囲において、一つのネットワークを構成している。例えば、各機器間をそれぞれ一本の1394シリアルバス用ケーブルでディジーチェーン接続するだけで、それぞれの機器が中継の役割を担うので、全体として一つのネットワークを構成することができる。

【 0 0 2 0 】

また、1394シリアルバスはPlug and Play機能に対応し、1394シリアルバス用ケーブルを機器に接続するだけで自動的に機器を認識し、接続状況を認識する機能を有している。また、図3に示すようなシステムにおいて、ネットワークからある機器が外されたり、または新たに加えられたときなど、自動的にバスをリセット（それまでのネットワークの構成情報をリセット）して、新たなネットワークを再構築する。この機能によって、その時々ネットワークの構成を常時設定、認識することができる。

【 0 0 2 1 】

また、1394シリアルバスのデータ転送速度は、100/200/400Mbpsが定義されていて、上位の転送速度をもつ機器が下位の転送速度をサポートすることで、互換性が保たれている。データ転送モードとしては、コントロール信号などの非同期

データを転送する非同期(Asynchronous)転送モード(ATM)と、リアルタイムなAVデータ等の同期データを転送する同期(Isochronous)転送モードがある。この非同期データと同期データは、各サイクル(通常125 μ s/サイクル)の中で、サイクル開始を示すサイクルスタートパケット(CSP)の転送に続き、同期データの転送を優先しつつ、サイクル内で混在して転送される。

【 0 0 2 2 】

図4は1394シリアルバスの構成例を示す図である。1394シリアルバスはレイヤ構造で構成されている。図4に示すように、コネクタポート810には、1394シリアルバス用のケーブル813の先端のコネクタが接続される。コネクタポート810の上位には、ハードウェア部800で構成されるフィジカルレイヤ811とリンクレイヤ812がある。ハードウェア部800はインタフェース用チップで構成され、そのうちフィジカルレイヤ811は符号化やコネクション関連の制御等を行い、リンクレイヤ812はパケット転送やサイクルタイムの制御等を行う。

【 0 0 2 3 】

ファームウェア部801のトランザクションレイヤ814は、転送(トランザクション)すべきデータの管理を行い、Read、Write、Lockの命令を出す。ファームウェア部801のマネージメントレイヤ815は、1394シリアルバスに接続されている各機器の接続状況やIDの管理を行い、ネットワークの構成を管理する。上記のハードウェアとファームウェアまでが、1394シリアルバスの実質的な構成である。

【 0 0 2 4 】

また、ソフトウェア部802のアプリケーションレイヤ816は、利用されるソフトによって異なり、インタフェース上でどのようにしてデータを転送するかは、プリンタやAV/Cプロトコルなどのプロトコルによって定義される。

【 0 0 2 5 】

図5は1394シリアルバスにおけるアドレス空間の一例を示す図である。1394シリアルバスに接続された各機器(ノード)には必ずノードに固有の64ビットアドレスをもたせる。そして、このアドレスは機器のメモリに格納されていて、自分や相手のノードアドレスを常時認識することで、通信相手を指定したデータ通信を行うことができる。

【 0 0 2 6 】

1394シリアルバスのアドレッシングは、IEEE1212規格に準じた方式であり、アドレス設定は、最初の10ビットがバスの番号の指定用に、次の6ビットがノードIDの指定用に使われる。

【 0 0 2 7 】

それぞれの機器内で使用される48ビットのアドレスについても、20ビットと28ビットに分けられ、256Mバイト単位の構造をもって利用される。最初の20ビットのアドレス空間のうち0～0xFFFFDはメモリ空間、0xFFFFEはプライベート空間、0xFFFFFはレジスタ空間とそれぞれ呼ばれる。プライベート空間は機器内で自由に利用できるアドレスであり、レジスタ空間にはバスに接続された機器間で共通な情報が置かれ、各機器間のコミュニケーションに使われる。

【 0 0 2 8 】

レジスタ空間の、最初の512バイトにはCSRアーキテクチャのコアになるレジスタ（CSRコア）が、次の512バイトにはシリアルバスのレジスタが、その次の1024バイトにはコンフィグレーションROMが、残りはユニット空間で機器固有のレジスタが、それぞれ置かれる。

【 0 0 2 9 】

一般的には異種バスシステムの設計の簡略化のため、ノードは初期ユニット空間の最初の2048バイトだけを使うべきであり、この結果としてCSRコア、シリアルバスのレジスタ、コンフィグレーションROMおよびユニット空間の最初の2048バイトを合わせて4096バイトで構成することが望ましい。

【 0 0 3 0 】

以上が、1394シリアルバスの概要である。次に、1394シリアルバスの特徴をより詳細に説明する。

【 0 0 3 1 】

【1394シリアルバスの詳細】

[1394シリアルバスの電氣的仕様]

図6は1394シリアルバス用のケーブルの断面を示す図である。1394シリアルバス用ケーブルには、二組のツイストペア信号線の他に、電源ラインが設けられて

いる。これによって、電源を持たない機器や、故障などにより電圧が低下した機器にも電力の供給が可能になる。電源線により供給される直流電力の電圧は8~40V、電流は最大電流1.5Aに規定されている。なお、DVケーブルと呼ばれる規格では、電源ラインを省いた四線で構成される。

【 0 0 3 2 】

[DS-Link方式]

図7は1394シリアルバスで採用されている、データ転送方式のDS-Link(Data/Strobe Link)方式を説明するための図である。

【 0 0 3 3 】

DS-Link方式は、高速なシリアルデータ通信に適し、二組の信号線を必要とする。つまり、二組のより対線のうち一組でデータ信号を送り、もう一組でストロブ信号を送る構成になっている。受信側では、このデータ信号と、ストロブ信号との排他的論理和をとることによってクロックを生成することができるという特徴がある。このため、DS-Link方式を用いると、データ信号中にクロック信号を混入させる必要がないので他のシリアルデータ転送方式に比べて転送効率が、高い、クロック信号を生成できるので位相ロックドロープ(PLL)回路が不要になり、その分コントローラLSIの回路規模を小さくすることができる。さらに、転送すべきデータが無いときにアイドル状態であることを示す情報を送る必要がないので、各機器のトランシーバ回路をスリープ状態にすることができ、消費電力の低減が図れる、などが挙げられる。

【 0 0 3 4 】

[バスリセットのシーケンス]

1394シリアルバスに接続されている各機器(ノード)にはノードIDが与えられ、ネットワークを構成するノードとして認識される。例えば、ネットワーク機器の接続分離や電源のオン/オフなどによるノード数の増減、つまりネットワーク構成に変化があり、新たなネットワーク構成を認識する必要があるとき、その変化を検知した各ノードはバス上にバスリセット信号を送信して、新たなネットワーク構成を認識するモードに入る。このネットワーク構成の変化の検知は、コネクタポート810においてバイアス電圧の変化を検知することによって行われる。

【0035】

あるノードからバスリセット信号が送信されると、各ノードのフィジカルレイヤ811はこのバスリセット信号を受けると同時にリンクレイヤ812にバスリセットの発生を伝達し、かつ他のノードにバスリセット信号を伝達する。最終的にすべてのノードがバスリセット信号を受信した後、バスリセットのシーケンスが起動される。なお、バスリセットのシーケンスは、ケーブルが抜き差しされた場合や、ネットワークの異常等をハードウェアが検出した場合に起動されるとともに、プロトコルによるホスト制御などフィジカルレイヤ811に直接命令を与えることによっても起動される。また、バスリセットのシーケンスが起動されると、データ転送は一時中断されてバスリセットの間は待たされ、バスリセット終了後、新しいネットワーク構成の基で再開される。

【0036】

[ノードID決定のシーケンス]

バスリセットの後、各ノードは新しいネットワーク構成を構築するために、各ノードにIDを与える動作に入る。このときの、バスリセットからノードID決定までの一般的なシーケンスを図8から図10に示すフローチャートを用いて説明する。図8は、バスリセット信号の発生から、ノードIDが決定し、データ転送が行えるようになるまでの一連のシーケンス例を示すフローチャートである。各ノードは、ステップS101でバスリセット信号の発生を常時監視し、バスリセット信号が発生するとステップS102に移り、ネットワーク構成がリセットされた状態において新たなネットワーク構成を得るために、互いに直結されているノード間で親子関係が宣言される。そしてステップS103の判定により、すべてのノード間で親子関係が決ったと判定されるまでステップS102が繰り返される。

【0037】

親子関係が決定するとステップS104へ進みルート(root)ノードが決定され、ステップS105で各ノードにIDを与えるノードIDの設定作業が行われる。ルートノードから所定のノード順にノードIDの設定が行われ、ステップS106の判定により、すべてのノードにIDが与えられたと判定されるまでステップS105が繰り返される。

【 0 0 3 8 】

ノードIDの設定が終了すると、新しいネットワーク構成がすべてのノードにおいて認識されたことになるのでノード間のデータ転送が行える状態になり、ステップS107でデータ転送が開始されるとともに、シーケンスはステップS101へ戻り、再びバスリセット信号の発生が監視される。

【 0 0 3 9 】

図9はバスリセット信号の監視(S101)からルートノードの決定(S104)までの詳細例を示すフローチャート、図10はノードID設定(S105,S106)の詳細例を示すフローチャートである。

【 0 0 4 0 】

図9において、ステップS201でバスリセット信号の発生が監視され、バスリセット信号が発生すると、ネットワーク構成は一旦リセットされる。次に、ステップS202で、リセットされたネットワーク構成を再認識する作業の第一歩として、各機器はフラグFLをリーフノードであることを示すデータでリセットする。そして、ステップS203で、各機器はポート数、つまり自分に接続されている他ノードの数を調べ、ステップS204で、ステップS203の結果に応じて、これから親子関係の宣言を始めるために、未定義（親子関係が決定されていない）ポートの数を調べる。ここで、未定義ポート数は、バスリセットの直後はポート数に等しいが、親子関係が決定されて行くにしたがって、ステップS204で検知される未定義ポートの数は減少する。

【 0 0 4 1 】

バスリセットの直後に親子関係の宣言を行えるのは実際のリーフノードに限られている。リーフノードであるか否かはステップS203のポート数の確認結果から知ることができ、つまりポート数が「1」であればリーフノードである。リーフノードは、ステップS205で、接続相手のノードに対して親子関係の宣言「自分は子、相手は親」を行い動作を終了する。

【 0 0 4 2 】

一方、ステップS203でポート数が「2以上」であったノード、つまりブランチノードは、バスリセットの直後は「未定義ポート数>1」であるからステップS20

6へ進み、フラグFLにブランチノードを示すデータをセットし、ステップS207で他ノードから親子関係が宣言されるのを待つ。他ノードから親子関係が宣言され、それを受けたブランチノードはステップS204に戻り、未定義ポート数を確認するが、もし未定義ポート数が「1」になっていれば残るポートに接続された他ノードに対して、ステップS205で「自分は子、相手は親」の親子関係を宣言することができる。また、未だ未定義ポート数が「2以上」あるブランチノードは、ステップS207で再び他ノードから親子関係が宣言されるのを待つことになる。

【0043】

何れか一つのブランチノード（または例外的に、子宣言を行えるのにもかかわらず、すばやく動作しなかったリーフノード）の未定義ポート数が「0」になると、ネットワーク全体の親子関係の宣言が終了したことになり、未定義ポート数が「0」になった唯一のノード、つまりすべてノードの親に決まったノードは、ステップS208でフラグFLにルートノードを示すデータをセットし、ステップS209でルートノードとして認識される。

【0044】

このようにして、バスリセットから、ネットワーク内のすべてのノード間における親子関係の宣言までの手順が終了する。

【0045】

次に、各ノードにIDを与える手順を説明するが、最初にIDの設定を行うことができるのはリーフノードである。そして、リーフ→ブランチ→ルートの順に若い番号（ノード番号：0）からIDを設定する。

【0046】

図10のステップS301で、フラグFLに設定されたデータを基にノードの種類、つまりリーフ、ブランチおよびルートに応じた処理に分岐する。

【0047】

まずリーフノードの場合は、ステップS302でネットワーク内に存在するリーフノードの数（自然数）を変数Nに設定した後、ステップS303で各リーフノードがルートノードに対して、ノード番号を要求する。この要求が複数ある場合、ルートノードはステップS304でアービトレーションを行い、ステップS305である一つ

のノードにノード番号を与え、他のノードにはノード番号の取得失敗を示す結果を通知する。

【 0 0 4 8 】

ステップS306の判断により、ノード番号を取得できなかったリーフノードは、再びステップS303でノード番号の要求を繰り返す。一方、ノード番号を取得できたリーフノードは、ステップS307で、取得したノード番号を含むID情報をブロードキャストすることで全ノードに通知する。ID情報のブロードキャストが終わるとステップS308で、リーフ数を表す変数Nがデクリメントされる。そして、ステップS309の判定により変数Nが「0」になるまでステップS303からS308の手順が繰り返され、すべてのリーフノードのID情報がブロードキャストされた後、ステップS310へ進んで、ブランチノードのID設定に移る。

【 0 0 4 9 】

ブランチノードのID設定もリーフノードとほぼ同様に行われる。まず、ステップS310でネットワーク内に存在するブランチノードの数（自然数）を変数Mに設定した後、ステップS311で各ブランチノードがルートノードに対して、ノード番号を要求する。この要求に対してルートノードは、ステップS312でアービトレーションを行い、ステップS313である一つのブランチノードにリーフノードに続く若い番号を与え、ノード番号を取得できなかったブランチノードには取得失敗を示す結果を通知する。

【 0 0 5 0 】

ステップS314の判定により、ノード番号の取得に失敗したことを知ったブランチノードは、再びステップS311でノード番号の要求を繰り返す。一方、ノード番号を取得できたブランチノードはステップS315で、取得したノード番号を含むID情報をブロードキャストすることで全ノードに通知する。ID情報のブロードキャストが終わるとステップS316で、ブランチ数を表す変数Mがデクリメントされる。そして、ステップS317の判定により、変数Mが「0」になるまでステップS311からS316の手順が繰返され、すべてのブランチノードのID情報がブロードキャストされた後、ステップS318へ進んで、ルートノードのID設定に移る。

【 0 0 5 1 】

ここまで終了すると、最終的にIDを取得していないノードはルートノードのみなので、ステップS318では、他のノードに与えていない最も若い番号を自分のノード番号に設定し、ステップS319でルートノードのID情報をブロードキャストする。

【 0 0 5 2 】

以上で、すべてのノードのIDが設定されるまでの手順が終了する。次に、図11に示すネットワーク例を用いてノードID決定のシーケンスの具体的な手順を説明する。

【 0 0 5 3 】

図11に示すネットワークは、ルートであるノードBの下位にはノードAとノードCが直結され、ノードCの下位にはノードDが直結され、ノードDの下位にはノードEとノードFが直結された階層構造を有する。この、階層構造やルートノード、ノードIDを決定する手順は以下のようになる。

【 0 0 5 4 】

バスリセットが発生した後、各ノードの接続状況を認識するために、各ノードの直結されているポート間において、親子関係の宣言がなされる。ここでいう親子とは、階層構造の上位が「親」、下位が「子」という意味である。図11では、バスリセットの後、最初に親子関係を宣言したのはノードAである。前述したように、一つのポートだけが接続されたノード（リーフ）から親子関係の宣言を開始することができる。これは、ポート数が「1」であればネットワークツリーの末端、つまりリーフノードであることが認識され、それらリーフノードの中で最も早く動作を行ったノードから親子関係が決定されて行くことになる。こうして親子関係の宣言を行ったノードのポートが、互いに接続された二つのノードの「子」と設定され、相手ノードのノードが「親」と設定される。こうして、ノードA-B間、ノードE-D間、ノードF-D間で「子-親」の関係が設定される。

【 0 0 5 5 】

さらに、階層が一つ上がって、複数のポートをもつノード、つまりブランチノードのうち他ノードから親子関係の宣言を受けたノードから順次、上位のノードに対して親子関係を宣言する。図11ではまずノードD-E間、D-F間の親子関係が決

定された後、ノードDがノードCに対して親子関係を宣言し、その結果、ノードD-C間で「子-親」の関係が設定される。ノードDから親子関係の宣言を受けたノードCは、もう一つのポートに接続されているノードBに対して親子関係を宣言し、これによってノードC-B間で「子-親」の関係が設定される。

【0056】

このようにして、図11に示すような階層構造が構成され、最終的に接続されているすべてのポートにおいて親となったノードBが、ルートノードと決定される。なお、ルートノードは一つのネットワーク構成中に一つしか存在しない。また、ノードAから親子関係を宣言されたノードBが、速やかに、他のノードに対して親子関係を宣言した場合は、例えばノードCなどの他のノードがルートノードになる可能性もあり得る。すなわち、親子関係の宣言が伝達されるタイミングによっては、どのノードもルートノードになる可能性があり、ネットワーク構成が同一であっても、特定のノードがルートノードになるとは限らない。

【0057】

ルートノードが決定されると、各ノードIDの決定モードに入る。すべてのノードは、決定した自分のID情報を、他のすべてのノードに通知するブロードキャスト機能をもっている。なお、ID情報は、ノード番号、接続されている位置の情報、もっているポートの数、接続のあるポートの数、各ポートの親子関係の情報などを含むID情報としてブロードキャストされる。

【0058】

ノード番号の割当ては、前述したようにリーフノードから開始され、順に、ノード番号=0,1,2,...が割当てられる。そしてID情報のブロードキャストによって、そのノード番号は割当て済みであることが認識される。

【0059】

すべてのリーフノードがノード番号を取得し終わると、次はブランチノードへ移りリーフノードに続くノード番号が割当てられる。リーフノードと同様に、ノード番号が割当てられたブランチノードから順にID情報がブロードキャストされ、最後にルートノードが自己のID情報をブロードキャストする。従って、ルートノードは常に最大のノード番号を所有することになる。

【 0 0 6 0 】

以上のようにして、階層構造全体のID設定が終わり、ネットワーク構成が構築され、バスの初期化作業が完了する。

〔ノード管理のための制御情報〕

ノード管理を行うためのCSRアーキテクチャの基本的な機能として、図5に示したCSRコアがレジスタ上に存在する。それらレジスタの位置と機能を図12に示すが、図中のオフセットは0xFFFFF0000000からの相対位置である。

【 0 0 6 1 】

CSRアーキテクチャでは、0xFFFFF0000200からシリアルバスに関するレジスタが配置されている。それらのレジスタの位置と機能を図13に示す。

【 0 0 6 2 】

また、0xFFFFF0000800から始まる場所には、シリアルバスのノード資源に関する情報が配置されている。それらのレジスタの位置と機能を図14に示す。

【 0 0 6 3 】

CSRアーキテクチャでは、各ノードの機能を表すためコンフィグレーションROMをもっているが、このROMには最小形式と一般形式があり、0xFFFFF0000400から配置される。最小形式では図15に示すようにベンダIDを表すだけであり、このベンダIDは24ビットで表される全世界で固有の値である。

【 0 0 6 4 】

また、一般形式は図16に示すような形式で、ノードに関する情報をもっているが、この場合、ベンダIDはルートディレクトリ(root_directory)にもつことができる。また、バス情報ブロック(bus info block)とルートリーフ(root leaf)にはベンダIDを含む64ビットの全世界で固有な装置番号をもっている。この装置番号は、バスリセットなどの再構成後に継続してノードを認識するために使用される。

〔シリアルバス管理〕

1394シリアルバスのプロトコルは、図4に示したように、フィジカルレイヤ811、リンクレイヤ812およびトランザクションレイヤ814から構成されている。この中で、バス管理は、CSRアーキテクチャに基づくノードの制御とバス資源管理の

ための基本的な機能を提供している。

【 0 0 6 5 】

バス管理を行うノード（以下「バス管理ノード」と呼ぶ）は、同一バス上に唯一存在し、シリアルバス上の他のノードに管理機能を提供するが、この管理機能にはサイクルマスタの制御や、性能の最適化、電源管理、伝送速度管理、構成管理などがある。

【 0 0 6 6 】

バス管理機能は、バスマネージャ、同期（アイソクロノス）リソースマネージャおよびノード制御の三つの機能に大きく分けられる。ノード制御は、CSRによってフィジカルレイヤ811、リンクレイヤ812、トランザクションレイヤ814およびアプリケーションにおけるノード間通信を可能にする管理機能である。同期リソースマネージャは、シリアルバス上で同期型のデータ転送を行うために必要になる管理機能で、同期データの転送帯域幅とチャネル番号の割当てを管理するものである。この管理を行うためにバス管理ノードは、バスの初期化後に、同期リソースマネージャ機能をもつノードの中から動的に選出される。

【 0 0 6 7 】

また、バス上にバス管理ノードが存在しない構成では、電源管理やサイクルマスタの制御のようなバス管理の一部の機能を同期リソースマネージャ機能をもつノードが行う。さらにバスマネージャは、アプリケーションに対してバス制御のインタフェイスを提供するサービスを行う管理機能であり、その制御インタフェイスにはシリアルバス制御要求(SB_CONTROL.request)、シリアルバスイベント制御確認(SB_CONTROL.confirmation)、シリアルバスイベント通知(SB_EVENT.indication)がある。

【 0 0 6 8 】

シリアルバス制御要求は、バスのリセット、バスの初期化、バスの状態情報などを、アプリケーションからバス管理ノードに要求する場合に利用される。シリアルバスイベント制御確認は、シリアルバス制御要求の結果で、バス管理ノードからアプリケーションに確認通知される。シリアルバスイベント通知は、バス管理ノードからアプリケーションに対して、非同期に発生されるイベントを通知す

るためのものである。

[データ転送プロトコル]

1394シリアルバスのデータ転送は、周期的に送信する必要のある同期データ（同期パケット）と、任意タイミングのデータ送受信が許容される非同期データ（非同期パケット）とが同時に存在し、なおかつ、同期データのリアルタイム性を保証している。データ転送では、転送に先立ってバス使用权を要求し、バスの使用許可を得るためのバスアービトレーションが行われる。

【0069】

非同期転送においては、送信ノードIDおよび受信ノードIDが転送データと一緒にパケットデータとして送られる。受信ノードは、自分のノードIDを確認してパケットを受取るとアクノリッジ信号を送信ノードに返すことで、一つのトランザクションが完了する。

【0070】

同期転送においては、送信ノードが伝送速度とともに同期チャネルを要求し、チャネルIDが転送データと一緒にパケットデータとして送られる。受信ノードは、所望するチャネルIDを確認してデータパケットを受取る。必要になるチャネル数と伝送速度はアプリケーションレイヤ816で決定される。

【0071】

これらのデータ転送プロトコルは、フィジカルレイヤ811、リンクレイヤ812およびトランザクションレイヤ814の三つのレイヤによって定義される。フィジカルレイヤ811は、バスとの物理的・電氣的インタフェース、ノード接続の自動認識、ノード間のバス使用权のバスアービトレーションなどを行う。リンクレイヤ812は、アドレッシング、データチェック、パケット送受信、そして同期転送のためのサイクル制御を行う。トランザクションレイヤ814は、非同期データに関する処理を行う。以下、各レイヤにおける処理について説明する。

[フィジカルレイヤ]

次に、フィジカルレイヤ811におけるバスアービトレーションを説明する。

【0072】

1394シリアルバスは、データ転送に先立って、必ず、バス使用权のアービトレ

ーションを行う。1394シリアルバスに接続された各機器は、ネットワーク上を転送される信号をそれぞれ中継することによって、ネットワーク内のすべての機器に同信号を伝える論理的なバス型ネットワークを構成するので、パケットの衝突を防ぐ意味でバスアービトレーションが必要である。これによって、ある時間には、一つのノードだけが転送を行うことができる。

【 0 0 7 3 】

図17はバス使用権の要求を説明する図、図18はバス使用の許可を説明する図である。バスアービトレーションが始まると、一つもしくは複数のノードが親ノードに向かって、それぞれバスの使用権を要求する。図17においては、ノードCとノードFがバス使用権を要求している。この要求を受けた親ノード（図17ではノードA）は、さらに親ノードに向かって、バスの使用権を要求することで、ノードFによるバスの使用権の要求を中継する。この要求は最終的に、アービトレーションを行うルートノードに届けられる。

【 0 0 7 4 】

バスの使用権の要求を受けたルートノードは、どのノードにバスの使用権を与えるかを決める。このアービトレーション作業はルートノードのみが行えるものであり、アービトレーションに勝ったノードにはバスの使用許可が与えられる。図18は、ノードCにバスの使用許可が与えられ、ノードFのバスの使用権の要求は拒否された状態を示している。

【 0 0 7 5 】

ルートノードは、バスアービトレーションに負けたノードに対してはDP(data prefix)パケットを送り、そのバスの使用権の要求が拒否されたことを知らせる。バスアービトレーションに負けたノードのバスの使用権の要求は、次回のバスアービトレーションまで待たされることになる。

【 0 0 7 6 】

以上のようにして、アービトレーションに勝ってバス使用の許可を得たノードは、以降、データ転送を開始することができる。ここで、バスアービトレーションの一連の流れを図19に示すフローチャートにより説明する。

【 0 0 7 7 】

ノードがデータ転送を開始できるためには、バスがアイドル状態であることが必要である。先に開始されたデータ転送が終了し、現在、バスがアイドル状態にあることを確認するためには、各転送モードで個別に設定されている所定のアイドル時間ギャップ長（例えば、サブアクションギャップ）の経過を検出し、所定のギャップ長が検出された場合、各ノードはバスがアイドル状態になったと判断する。各ノードは、ステップS401で、非同期データ、同期データなどそれぞれ転送するデータに応じた所定のギャップ長が検出されたか否かを判断する。所定のギャップ長が検出されない限り、転送を開始するために必要なバス使用权を要求することはできない。

【 0 0 7 8 】

各ノードは、ステップS401で所定のギャップ長が検出されると、ステップS402で転送すべきデータがあるか判断し、ある場合はステップS403でバスの使用权を要求する信号をルートに対して発信する。このバスの使用权の要求を表す信号は、図17に示すように、ネットワーク内の各機器に中継されながら、最終的にルートノードに届けられる。ステップS402で転送するデータがないと判断した場合は、ステップS401に戻る。

【 0 0 7 9 】

ルートノードは、ステップS404でバスの使用权を要求する信号を一つ以上受信したら、ステップS405で使用权を要求したノードの数を調べる。ステップS405の判定により、使用权を要求したノードが一つだったら、そのノードに、直後のバス使用許可が与えられることになる。また、使用权を要求したノードが複数だったら、ステップS406で直後のバス使用許可を与えるノードを一つに絞るアービトレーション作業が行われる。このアービトレーション作業は、毎回同じノードばかりにバスの使用許可を与えるようなことはなく、平等にバスの使用許可を与えるようになっている（フェア・アービトレーション）。

【 0 0 8 0 】

ルートノードの処理は、ステップS407で、ステップS406のアービトレーションに勝った一つのノードと、敗れたその他のノードとに応じて分岐する。アービトレーションに勝った一つのノード、またはバスの使用权を要求したノードが一つ

の場合は、ステップS408でそのノードに対してバスの使用許可を示す許可号が送られる。この許可信号を受信したノードは、直後に転送すべきデータ（パケット）の転送を開始する（ステップS410）。また、アービトレーションに敗れたノードにはステップS409で、バス使用権の要求が拒否されたことを示すDP(data prefix)パケットが送られる。DPパケットを受取ったノードの処理は、再度、バスの使用権を要求するためにステップS401まで戻る。ステップS410におけるデータの転送が完了したノードの処理もステップS401へ戻る。

【トランザクションレイヤ】

トランザクションの種類には、リードトランザクション、ライトトランザクションおよびロックトランザクションの三種類がある。

【0081】

リードトランザクションでは、イニシエータ（要求ノード）がターゲット（レスポンスノード）のメモリの特定アドレスからデータを読取る。ライトトランザクションでは、イニシエータがターゲットのメモリの特定アドレスにデータを書込む。また、ロックトランザクションでは、イニシエータからターゲットに参照データと更新データを転送する。その参照データは、ターゲットのアドレスのデータと組み合わせられて、ターゲットの特定のアドレスを指示する指定アドレスになる。そして、この指定アドレスのデータが更新データにより更新される。

【0082】

図20はトランザクションレイヤ814におけるCSRアーキテクチャに基づくリード、ライト、ロックの各コマンドの要求・レスポンスプロトコルを示す図で、図に示す要求、通知、レスポンスおよび確認は、トランザクションレイヤ814でのサービス単位である。

【0083】

トランザクション要求(TR_DATA.request)はレスポンスノードに対するパケットの転送、トランザクション通知(TR_DATA.indication)はレスポンスノードに要求が届いたことの通知、トランザクションレスポンス(TR_DATA.response)はアクノリッジの送信、トランザクション確認(TR_DATA.confirmation)はアクノリッジの受信である。

[リンクレイヤ]

図21はリンクレイヤ812におけるサービスを示す図で、レスポンスノードに対するパケットの転送を要求するリンク要求(LK_DATA.request)、レスポンスノードにパケット受信を通知するリンク通知(LK_DATA.indication)、レスポンスノードからのアクノリッジ送信のリンクレスポンス(LK_DATA.response)、要求ノードのアクノリッジ送信のリンク確認(LK_DATA.confirmation)のサービス単位に分けられる。

【0084】

一つのパケット転送プロセスはサブアクションと呼ばれ、非同期サブアクションと同期サブアクションの二つの種類がある。以下では、各サブアクションの動作について説明する。

[非同期サブアクション]

非同期サブアクションは非同期データ転送である。図22は非同期転送における時間的な遷移を示す図である。図22に示す最初のサブアクションギャップは、バスのアイドル状態を示すものである。このアイドル時間が所定値になった時点で、データ転送を希望するノードがバス使用权を要求し、バスアービトレーションが実行される。

【0085】

バスアービトレーションによりバスの使用が許可されると、次に、データがパケット転送され、このデータを受信したノードは、ACKギャップという短いギャップの後、受信確認用返送コードACKを返してレスポンスするか、レスポンスパケットを返送することでデータ転送が完了する。ACKは4ビットの情報と4ビットのチェックサムからなり、成功、ビジー状態またはペンディング状態であることを示す情報を含み、すぐにデータ送信元のノードに返される。

【0086】

図23は非同期転送用パケットのフォーマットを示す図である。パケットには、データ部および誤り訂正用のデータCRCのほかにヘッダ部があり、そのヘッダ部には目的ノードID、ソースノードID、転送データ長や各種コードなどが書込まれている。

【 0 0 8 7 】

また、非同期転送は送信ノードから受信ノードへの一対一の通信である。送信元ノードから送り出されたパケットは、ネットワーク中の各ノードに行き渡るが、各ノードは自分宛てのパケット以外は無視するので、宛先に指定されたノードだけがそのパケットを受取ることになる。

〔同期サブアクション〕

1394シリアルバスの最大の特徴であるともいえるこの同期転送は、とくにAVデータなどのリアルタイム転送を必要とするデータの転送に適している。また、非同期転送が一対一の転送であるのに対し、この非同期転送はブロードキャスト機能によって、一つの送信元ノードから他のすべてのノードへ様にデータを転送することができる。

【 0 0 8 8 】

図24は同期転送における時間的な遷移を示す図で、同期転送はバス上で一定時間毎に実行され、この時間間隔を同期サイクルと呼ぶ。同期サイクル時間は125 μ sである。この同期サイクルの開始を示し、各ノードの動作を同期させる役割を担っているのがサイクルスタートパケット(CSP)2000である。CSP2000を送信するのは、サイクルマスタと呼ばれるノードであり、一つ前のサイクル内の転送が終了し、所定のアイドル期間（サブアクションギャップ2001）を経た後、本サイクルの開始を告げるCSP2000を送信する。つまり、このCSP2000が送信される時間間隔が125 μ sになる。

【 0 0 8 9 】

また、図24にチャンネルA、チャンネルBおよびチャンネルCと示すように、一つの同期サイクル内において複数種のパケットにチャンネルIDをそれぞれ与えることにより、それらのパケットを区別して転送することができる。これにより、複数ノード間で、略同時に、リアルタイム転送が可能であり、また、受信ノードは所望するチャンネルIDのデータのみを受信すればよい。このチャンネルIDは、受信ノードのアドレスなどを表すものではなく、データに対する論理的な番号に過ぎない。従って、送信されたあるパケットは、一つの送信元ノードから他のすべてのノードに行き渡る、つまりブロードキャストされることになる。

【 0 0 9 0 】

同期転送によるパケット送信に先立ち、非同期転送と同様に、バスアービトレーションが行われる。しかし、非同期転送のように一對一の通信ではないので、同期転送には受信確認用の返送コードのACKは存在しない。

【 0 0 9 1 】

また、図24に示したisoギャップ（同期ギャップ）は、同期転送を行う前にバスがアイドル状態であることを確認するために必要なアイドル期間を表している。この所定のアイドル期間を検出したノードは、バスがアイドル状態にあると判断し、同期転送を行いたい場合はバス使用权を要求するのでバスアービトレーションが行われることになる。

【 0 0 9 2 】

図25は同期転送用のパケットフォーマット例を示す図である。各チャンネルに分けられた各種のパケットには、それぞれデータ部および誤り訂正用のデータCRCのほかにヘッダ部があり、そのヘッダ部には図26に示すような、転送データ長、チャンネル番号、その他各種コードおよび誤り訂正用のヘッダCRCなどが書込まれている。

〔バス・サイクル〕

実際に、1394シリアルバスにおいては、同期転送と非同期転送が混在できる。図27は同期転送と非同期転送が混在するときの転送状態の時間的遷移を示す図である。

【 0 0 9 3 】

ここで、前述したように同期転送は非同期転送より優先して実行される。その理由は、CSPの後、非同期転送を起動するために必要なアイドル期間のギャップ（サブアクションギャップ）よりも短いギャップ（アイソクロナスギャップ）で、同期転送を起動できるからである。従って、非同期転送より同期転送は優先して実行されることになる。

【 0 0 9 4 】

図27に示す一般的なバスサイクルにおいて、サイクル#mのスタート時にCSPがサイクルマスタから各ノードに転送される。CSPによって、各ノードの動作が同

期され、所定のアイドル期間（同期ギャップ）を待ってから同期転送を行おうとするノードはバスアービトレーションに参加し、パケット転送に入る。図27ではチャンネルe、チャンネルsおよびチャンネルkが順に同期転送されている。

【0095】

このバスアービトレーションからパケット転送までの動作を、与えられているチャンネル分繰り返し行った後、サイクル#mにおける同期転送がすべて終了すると、非同期転送を行うことができるようになる。つまり、アイドル時間が、非同期転送が可能なサブアクションギャップに達することによって、非同期転送を行いたいノードはバスアービトレーションに参加する。ただし、非同期転送が行えるのは、同期転送の終了から、次のCSPを転送すべき時間(cycle synch)までの間に、非同期転送を起動するためのサブアクションギャップが検出された場合に限られる。

【0096】

図27に示すサイクル#mでは、三つのチャンネル分の同期転送の後、非同期転送によりACKを含む2パケット（パケット1、パケット2）が転送されている。この非同期パケット2の後、サイクルm+1をスタートすべき時間(cycle synch)に至るので、サイクル#mにおける転送はこれで終わる。ただし、非同期または同期転送中に次のCSPを送信すべき時間(cycle synch)に至ったら、転送を無理に中断せず、その転送が終了した後にアイドル期間を経て次の同期サイクルのCSPを送信する。すなわち、一つの同期サイクルが125 μ s以上続いたときは、その延長分、次の同期サイクルは基準の125 μ sより短縮される。このように同期サイクルは125 μ sを基準に超過、短縮し得るものである。

しかし、同期転送はリアルタイム転送を維持するために、必要であれば毎サイクル実行され、非同期転送は同期サイクル時間が短縮されたことによって次以降の同期サイクルに延期されることもある。サイクルマスタは、こういった遅延情報も管理する。

【0097】

【画像処理分担】

上述した図1に示すようなシステム構成において、デジタルカメラ1とプリンタ

2とを1対1に接続する際に、上述した構成及び機能を有する1394シリアルバスを使用することで、より高速なデータ通信が可能になる。尚、本実施形態では、デジタルカメラ1とプリンタ2間において、非同期転送を行うとする。

【0098】

一般に、パーソナルコンピュータ（以降、PC）とプリンタとを接続したシステムにおいて印刷処理を行う際には、PCは印刷処理を早く終了して、速やかに他の作業を遂行することが要求される。従って、該システムにおいて印刷処理のスループットを向上させることを考えた場合、まず、PCからプリンタへのデータ転送時間の短縮が要求される。そして、次にプリンタ側における処理及び動作の高速化、更にプリンタバッファの高容量化、等が要求される。

【0099】

一方、本実施形態におけるデジタルカメラとプリンタとを接続したシステムにおいては、トータルスループットの向上のために、まずプリンタ側における印刷処理時間の短縮が要求される、

本実施形態においては、1394シリアルバスを利用することにより高速なデータ転送が可能であるため、一連の印刷処理のスループットを向上させるためには、デジタルカメラ1とプリンタ2における画像処理の分担が重要な要因となってくる。そこで本実施形態においては、デジタルカメラ1とプリンタ2とにおいて、圧縮データの伸長処理を適切に分担することを特徴とする。

【0100】

尚、本実施形態においては以下のことを前提として、説明を行う。即ち、1394シリアルバスによるデータ転送が十分に高速であるとする。また、プリンタ2における印刷動作が十分に高速であるため、画像処理の内容によってはプリンタ2側が処理待ち状態になることもあるとする。また、デジタルカメラ1側において、画像データはJPEG形式で圧縮保持されとする。

【0101】

以下、本実施形態における一連の印刷処理について詳細に説明する。

【0102】

本実施形態の画像処理システムにおいて、デジタルカメラ1において画像入力

部8で撮影された画像データは、画像処理部9で必要に応じて色補正、データ圧縮等の画像処理が施された後にRAM7へ格納される。そして、該圧縮データはI/F3へ渡されることにより、1394シリアルバスを介してプリンタ2へ転送される。

【 0 1 0 3 】

一方、I/F部4によって圧縮データを受け取ったプリンタ2では、該圧縮データをデータ処理部13へ送って伸長し、必要に応じてイメージデータ補正、色補正等を行った後、RGB-CMYK変換、ハーフトーン化、及びネイティブコマンド化を行ってプリンタ駆動部15へ送出することにより、記録用紙上への画像形成処理、即ち印刷動作が開始される。

【 0 1 0 4 】

図28A及び図28Bは、上述した本実施形態における一連の印刷処理の詳細を示すフローチャートであり、図28Aがデジタルカメラ1、図28Bがプリンタ2における処理を示す。

【 0 1 0 5 】

まず、図28Aに示すデジタルカメラ1側における処理について説明する。

【 0 1 0 6 】

ステップS3101で印刷処理が開始されると、ステップS3102で画像メモリより印刷対象となるJPEGデータを読み出し、ステップS3103で該JPEGデータを所定のブロック単位に分割する。そしてステップS3104において、I/F部3の状態を検出し、データ転送を可能とするレディ状態であればステップS3105へ進み、データ転送を不可とする待機状態（ビジー状態）であればステップS3108へ進む。ここで、I/F部3におけるレディ/ビジーの判定は、プリンタ2のプリントバッファに空きがあるか否かに基づいて行われる。即ち、プリンタ2のバッファが満杯であり、空きがなければビジー状態であると判定される。一般に、印刷処理の初期段階においてはプリンタ2のバッファには空きがあるため、データ転送は可能であり、従って最初はステップS3105へ処理が分岐する。

【 0 1 0 7 】

ステップS3105においては、JPEGデータを転送する旨をプリンタ2側へ通知し、ステップS3106でJPEGデータを1ブロック毎に転送する。そしてステップS3107で

は、印刷画像データがまだ残っていればステップS3104へ戻って処理を繰り返し、残りのデータが無ければ、デジタルカメラ1側における印刷のための処理を終了する。

【 0 1 0 8 】

一方、ステップS3104においてI/F状態がビジーであれば、ステップS3108で残りのJPEGデータを1ブロック毎に伸長し、ステップS3109でI/F状態がレディになったか否かをチェックする。未だビジーであれば、ステップS3108のJPEGブロック伸長を繰り返し、レディになればステップS3110へ進む。ステップS3110においては、JPEGデータを伸長したRAW-RGBデータを転送する旨をプリンタ2側へ通知し、ステップS3111でRAW-RGBデータを1ブロック毎に転送する。そしてステップS3112において印刷画像データがまだ残っていればステップS3108へ戻って処理を繰り返し、残りのデータが無ければ処理を終了する。

【 0 1 0 9 】

次に、図28Bに示すプリンタ2側における処理について説明する。

【 0 1 1 0 】

プリンタ2側において印刷処理が開始されると、まずステップS3113において、I/F部4を介して転送されてきた印刷画像データがRAM12内のバッファ領域にあるか否かをチェックする。印刷画像データがなければ転送されてくるまで待ち、あればステップS3114へ進む。

【 0 1 1 1 】

ステップS3114においては、プリンタ2内においてJPEGフラグがセットされていれば、該印刷画像データはJPEGデータであると判断してステップS3115へ進み、JPEGデータを伸長してステップS3116へ進む。

【 0 1 1 2 】

一方、ステップS3114においてJPEGフラグがセットされていなければ、該印刷画像データはRAW-RGBデータであると判断してステップS3116へそのまま進む。尚、プリンタ2のCPU10は、デジタルカメラ1からのJPEGデータの転送通知(ステップS3105)、またはRAW-RGBデータの転送通知(ステップS3110)を受けた時点で、JPEGフラグをセットまたはリセットする。即ち、転送されてくる印刷画像データがJP

EGデータであればJPEGフラグをセットし、RAW-RGBデータであればJPEGフラグをリセットする。

【0 1 1 3】

そしてステップS3116においては印刷に必要な所定の画像処理を行い、ステップS3117で印刷処理を実行して、一連の印刷処理が終了する。

【0 1 1 4】

以上説明したように本実施形態によれば、デジタルカメラ1側によるJPEGデータの転送中に、I/F部3においてプリンタ2内のバッファ状態を検知し、転送不可であるビジー状態となった時点で、残りのJPEGデータを伸長してRAW-RGBデータ転送に切り替える。

【0 1 1 5】

即ち、転送用のデータ形式の切り替えのタイミングをデジタルカメラ1側において判断し、JPEGデータの伸長処理を行うことにより、システムにおいて効率的な印刷処理が可能となり、トータルスループットの向上が図れる。

【0 1 1 6】

<第2実施形態>

以下、本発明に係る第2実施形態について説明する。

【0 1 1 7】

第2実施形態における画像処理システムは上述した第1実施形態と同様の構成からなり、やはりデジタルカメラ1とプリンタ2間において非同期転送を行うため、詳細な説明を省略する。

【0 1 1 8】

尚、第2実施形態においても、1394シリアルバスによるデータ転送は十分に高速であること、プリンタ2における印刷動作は十分に高速であるためプリンタ2側が処理待ち状態になることもあるということ、及び、デジタルカメラ1側において画像データはJPEG形式で圧縮保持されることを前提とする。

【0 1 1 9】

第2実施形態においても、デジタルカメラ1とプリンタ2とで圧縮データの伸長処理を適切に分担するが、分担のタイミングをプリンタ2側において提供するこ

とを特徴とする。

【 0 1 2 0 】

以下、第2実施形態における一連の印刷処理について詳細に説明する。

【 0 1 2 1 】

図29A及び図29Bは、第2実施形態における一連の印刷処理の詳細を示すフローチャートであり、図29Aがデジタルカメラ1、図29Bがプリンタ2における処理を示す。

【 0 1 2 2 】

まず、図28Aに示すデジタルカメラ1側における処理について説明する。

【 0 1 2 3 】

ステップS3201で印刷処理が開始されると、ステップS3202で画像メモリより印刷対象となるJPEGデータを読み出し、ステップS3203で該JPEGデータを所定のブロック単位に分割する。そしてステップS3204において、プリンタ2からのJPEGデータ伸長要求があるか否かを判定し、伸長要求があればステップS3207へ進み、なければステップS3205へ進む。一般に、印刷処理の初期段階であれば伸長要求の発行はないため、最初はステップS3205へ進む。

【 0 1 2 4 】

ステップS3205においては、JPEGデータを1ブロック毎に転送する。そしてステップS3206では、印刷画像データがまだ残っていればステップS3204へ戻って処理を繰り返し、残りのデータが無ければ、デジタルカメラ1側における印刷のための処理を終了する。

【 0 1 2 5 】

一方、ステップS3204においてプリンタ2からのJPEGデータ伸長要求があれば、ステップS3207で残りのJPEGデータを1ブロック毎に伸長してRAW-RGBデータを作成し、ステップS3208で該RAW-RGBデータを1ブロック毎に転送する。そしてステップS3209において印刷画像データがまだ残っていればステップS3207へ戻って処理を繰り返し、残りのデータが無ければ処理を終了する。

【 0 1 2 6 】

次に、図29Bに示すプリンタ2側における処理について説明する。

【 0 1 2 7 】

プリンタ2側において印刷処理が開始されると、まずステップS3210において、I/F部4を介して転送されてきた印刷画像データがRAM12内のバッファ領域にあるか否かをチェックする。印刷画像データがなければ転送されてくるまで待ち、あればステップS3211へ進む。

【 0 1 2 8 】

ステップS3211においては、バッファの空きが無くなった（バッファフル）か否かを判定し、空きがなければステップS3217へ、空きがあればステップS3212へ進む。尚、印刷処理の初期段階においてはバッファは空であるため、処理はステップS3212へ進むことになる。ステップS3212においては、デジタルカメラ1から転送されてきたJPEGデータをバッファから読み出し、ステップS3213で伸長する。そして、ステップS3214において印刷に必要な所定の画像処理を行い、ステップS3215で印刷処理を実行した後、ステップS3216で残りデータがあればステップS3211へ戻り、無ければ印刷処理を終了する。

【 0 1 2 9 】

一方、ステップS3211においてバッファフルが検出されると、ステップS3217においてデジタルカメラ1側へJPEGデータ伸長要求を発信する。

【 0 1 3 0 】

そしてステップS3218においては、デジタルカメラ1から転送されてきたJPEGデータをバッファから読み出し、ステップS3219で残りJPEGデータがあればステップS3220で伸長し、残りJPEGデータが無ければステップS3221へ進む。そして、ステップS3221において印刷に必要な所定の画像処理を行い、ステップS3222で印刷処理を実行した後、ステップS3223で残りデータがあればステップS3218へ戻り、無ければ印刷処理を終了する。

【 0 1 3 1 】

以上説明したように第2実施形態によれば、JPEGデータの転送中にプリンタ2は自身のバッファ状態を検知し、バッファフルとなった時点で、デジタルカメラ1側へJPEGデータの伸長要求を発信し、デジタルカメラ1側では該伸長要求に応じ、残りのJPEGデータを伸長してRAW-RGBデータ転送に切り替える。

【 0 1 3 2 】

即ち、転送用のデータ形式の切り替えのタイミングをプリンタ2側において判断し、デジタルカメラ1側へJPEGデータの伸長処理を要求することにより、システムにおいて効率的な印刷処理が可能となり、トータルスループットの向上が図れる。

【 0 1 3 3 】

<第3実施形態>

以下、本発明に係る第3実施形態について説明する。

【 0 1 3 4 】

第3実施形態における画像処理システムは上述した第1実施形態と同様の構成からなり、やはりデジタルカメラ1とプリンタ2間において非同期転送を行うため、詳細な説明を省略する。

【 0 1 3 5 】

尚、第3実施形態においても、1394シリアルバスによるデータ転送は十分に高速であること、プリンタ2における印刷動作は十分に高速であるためプリンタ2側が処理待ち状態になることもあるということ、及び、デジタルカメラ1側において画像データはJPEG形式で圧縮保持されることを前提とする。

【 0 1 3 6 】

第3実施形態においても、デジタルカメラ1とプリンタ2とで圧縮データの伸長処理を適切に分担するが、分担のタイミングをプリンタ2側において提供することを特徴とする。

【 0 1 3 7 】

以下、第3実施形態における一連の印刷処理について詳細に説明する。

【 0 1 3 8 】

図30A及び図30Bは、第3実施形態における一連の印刷処理の詳細を示すフローチャートであり、図30Aがデジタルカメラ1、図30Bがプリンタ2における処理を示す。

【 0 1 3 9 】

まず、図30Aに示すデジタルカメラ1側における処理について説明する。

【 0 1 4 0 】

ステップS3301で印刷処理が開始されると、ステップS3302で画像メモリより印刷対象となるJPEGデータを読み出し、ステップS3303で該JPEGデータを所定のブロック単位に分割する。そしてステップS3304において、プリンタ2によって発行された、JPEGデータ伸長要求を示す伸長要求フラグが受信されているか否かを判定し、受信されていればステップS3307へ進み、受信されていなければステップS3305へ進む。一般に、印刷処理の初期段階では伸長要求は発行されないため、最初はステップS3305へ進む。

【 0 1 4 1 】

ステップS3305においては、JPEGデータを転送する旨をプリンタ2側へ通知し、1ブロックのJPEGデータを転送する。そしてステップS3306では、印刷画像データがまだ残っていればステップS3304へ戻って処理を繰り返し、残りのデータが無ければ、デジタルカメラ1側における印刷のための処理を終了する。

【 0 1 4 2 】

一方、ステップS3304において伸長要求フラグがセットされていれば、ステップS3307で残りのJPEGデータの1ブロックを伸長してRAW-RGBデータを作成し、ステップS3308でJPEGデータを伸長したRAW-RGBデータを転送する旨をプリンタ2側へ通知して該1ブロックのRAW-RGBデータを転送した後、ステップS3306へ進む。

【 0 1 4 3 】

次に、図30Bに示すプリンタ2側における処理について説明する。

【 0 1 4 4 】

プリンタ2側において印刷処理が開始されると、まずステップS3309において、I/F部4を介して転送されてきた印刷画像データがRAM12内のバッファ領域にあるか否かをチェックする。印刷画像データがなければ転送されてくるまで待ち、あればステップS3310へ進む。

【 0 1 4 5 】

ステップS3310においては、バッファの空きが無くなった（バッファフル）か否かを判定し、空きがあればステップS3311へ進むが、空きが無ければステップS3318へ進み、JPEGデータ伸長要求フラグをセットしてデジタルカメラ1側へ送信

した後に、ステップS3311へ進む。尚、印刷処理の初期段階においてはバッファは空であるため、処理はそのままステップS3311へ進むことになる。

【 0 1 4 6 】

ステップS3311においては、バッファが空になった（バッファエンプティ）か否かを判定し、空でなければステップS3312へ進むが、空であればステップS3319へ進み、JPEGデータ伸長要求フラグをリセットしてデジタルカメラ1側へ送信した後に、ステップS3312へ進む。

【 0 1 4 7 】

そしてステップS3312においては、デジタルカメラ1から転送されてきたJPEGデータをバッファから読み出し、ステップS3313で残りJPEGデータがあればステップS3314で伸長し、残りJPEGデータが無ければステップS3315へ進む。そして、ステップS3315において印刷に必要な所定の画像処理を行い、ステップS3316で印刷処理を実行した後、ステップS3317で残りデータがあればステップS3310へ戻り、無ければ印刷処理を終了する。

【 0 1 4 8 】

以上説明したように第3実施形態によれば、JPEGデータの転送中にプリンタ2は自身のバッファ状況（フル/エンプティ）を検知し、該バッファ状況を示すフラグをその変化毎にデジタルカメラ1側へ逐次発信し、デジタルカメラ1側では該フラグに応じてプリンタ2側のバッファ情報を検知し、残りのJPEGデータを伸長するか否かをブロック毎に判断する。

【 0 1 4 9 】

即ち、デジタルカメラ1側において、プリンタ2から通知されたバッファ状況に基づいてJPEGデータの伸長処理を行うことにより、システムにおいて効率的な印刷処理が可能となり、トータルスループットの向上が図れる。

【 0 1 5 0 】

<第4実施形態>

以下、本発明に係る第4実施形態について説明する。

【 0 1 5 1 】

第4実施形態における画像処理システムは上述した第1実施形態と同様の構成

からなり、やはりデジタルカメラ1とプリンタ2間において非同期転送を行うため、詳細な説明を省略する。

【 0 1 5 2 】

尚、第4実施形態においては、1394シリアルバスによるデータ転送速度は十分ではないこと、及び、プリンタ2における印刷動作は十分に高速であるためにプリンタ2側が処理待ち状態になることもあること、デジタルカメラ1側において画像データはJPEG形式で圧縮保持されること、を前提とする。

【 0 1 5 3 】

上述した第1乃至第3実施形態においては、デジタルカメラ1に保持されているJPEGデータをプリンタ2に転送する際に、必要に応じてデジタルカメラ側で伸長した後に転送する例について説明したが、この場合、転送される最初のデータは、JPEGデータであった。

【 0 1 5 4 】

しかしながら、一般にデジタルカメラ1においては、撮影後に圧縮保持されたJPEGデータを印刷する前に表示を行うことも可能であり、この場合、表示処理の際にJPEGデータは一旦伸長される。従って、表示処理後に印刷を行う場合には、表示処理の際に伸長された画像データをそのままプリンタ2側へ転送することが考えられる。

【 0 1 5 5 】

しかしながら第4実施形態のようにデータ転送速度が十分に高速ではない場合には、転送すべきデータ量の増加に伴ってより長い転送時間が必要となるため、全体としてのスループットはむしろ低下してしまうことさえある。

【 0 1 5 6 】

そこで第4実施形態においては、デジタルカメラ1において既に伸長されたデータが存在していたとしても、印刷処理の際には元のJPEGデータをプリンタ2側へ転送する。

【 0 1 5 7 】

尚、その後の処理、即ち、最初のJPEGデータを転送した後は、上述した第1乃至第3実施形態と同様の伸長処理の分担制御を行なうことも可能であるが、デ

ータ転送速度が低速であれば、伸長処理はプリンタ2側のみに于行う方が全体の処理速度を向上させるためには望ましい。

【0158】

以上説明したように第4実施形態によれば、印刷処理の際に最初のデータ転送をJPEGデータをに限定することにより、転送時間が短縮できるため、プリンタ2側においてもより早いタイミングで印刷動作を開始することができる。従って、システムにおけるトータルスループットの向上が図れる。

【0159】

<第5実施形態>

以下、本発明に係る第5実施形態について説明する。

【0160】

第5実施形態における画像処理システムは上述した第1実施形態と同様の構成からなり、やはりデジタルカメラ1とプリンタ2間において非同期転送を行うため、詳細な説明を省略する。

【0161】

尚、第5実施形態においては、プリンタ2における印刷動作は十分に高速であるためにプリンタ2側が処理待ち状態になることもあること、及び、デジタルカメラ1側において画像データはJPEG形式で圧縮保持されること、を前提とする。

【0162】

第5実施形態においては、印刷対象となる最初の画像データが圧縮形式であった場合に、いずれの装置で伸長するのが最も効率的であるかを、デジタルカメラ1のCPU5の制御に基づいて以下のように判断することを特徴とする。

【0163】

まず、データ転送速度が十分に高速、即ち所定値以上であれば、データ転送時間が短くてすむため、JPEGデータの伸長処理はデジタルカメラ1側に行う方が効率的であると判断する。

【0164】

一方、データ転送速度が所定値以下であれば、デジタルカメラ1とプリンタ2の処理能力を比較して、処理能力の高い方で伸長処理を行うと判断する。尚、この

場合の処理能力の比較方法としては例えば、デジタルカメラ1をプリンタ2に接続する際に、プリンタ2におけるConfiguration ROMの内容を読み込み、そこに記載されている機種情報を参照することによって、プリンタ2の処理能力を把握することができる。また例えば、プリンタ2のConfiguration ROMがその機種情報を含んでいない場合には、プリンタ2側でデジタルカメラ1のConfiguration ROMを読み込んで互いの処理能力の判定を行い、その判定結果をデジタルカメラ1側に通知させることも可能である。

【0165】

また、データ転送速度が更に低速である場合には、伸長処理をプリンタ2側のみで行う方が効率的であると判断する。

【0166】

上述したように、データ転送速度に応じて最初のJPEGデータを伸長する装置を決定した後に、上述した第1乃至第3実施形態で説明した印刷処理を実行すればよい。

【0167】

以上説明したように第5実施形態によれば、印刷処理の際に最初のJPEGデータの伸長処理をデジタルカメラ1、プリンタ2のいずれで行った方が効率的であるかを適切に判断することにより、印刷処理のスループットを向上させることができる。

【0168】

尚、上述した各実施形態においてはIEEE1394に規定されるシリアルインタフェイスを用いてネットワークを構成する例を説明したが、本発明はこれに限定されるものではなく、Universal Serial Bus(USB)と呼ばれるシリアルインタフェイスなど、任意のシリアルインタフェイスを用いて構成されるネットワークにも適用することができる。

【0169】

【他の実施形態】

なお、本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど）から構成されるシステムに適用しても、一つの機器

からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

【0170】

また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体（または記録媒体）を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム(OS)などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0171】

さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張カードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張カードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0172】

本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明した図28乃至図30に示すフローチャートに対応するプログラムコードが格納されることになる。

【0173】

【発明の効果】

以上説明したように本発明によれば、画像入力装置と画像形成装置を接続した画像処理システムにおいて、転送データの形式を適切に変更することによって、

トータルスループットを向上させることができる。

【図面の簡単な説明】

【図1】

本発明を適用する画像処理システムの構成を示すブロック図、

【図2】

本実施形態におけるIEEE1394I/F部の詳細構成を示すブロック図、

【図3】

1394シリアルバスによるネットワークの構成例を示す図、

【図4】

1394シリアルバスの構成例を示す図、

【図5】

1394シリアルバスにおけるアドレス空間の一例を示す図、

【図6】

1394シリアルバス用のケーブルの断面を示す図、

【図7】

1394シリアルバスで採用されている、データ転送方式のDS-Link方式を説明するための図、

【図8】

バスリセット信号の発生から、ノードIDが決定し、データ転送が行えるようになるまでの一連のシーケンス例を示すフローチャート、

【図9】

バスリセット信号の監視からルートノードの決定までの詳細例を示すフローチャート、

【図10】

ノードID設定の詳細例を示すフローチャート、

【図11】

1394シリアルバスのネットワーク動作例を示す図、

【図12】

1394シリアルバスのCSRアーキテクチャの機能を示す図、

【図13】

1394シリアルバスに関するレジスタを示す図、

【図14】

1394シリアルバスのノード資源に関するレジスタを示す図、

【図15】

1394シリアルバスのコンフィギュレーションROMの最小形式を示す図、

【図16】

1394シリアルバスのコンフィギュレーションROMの一般形式を示す図、

【図17】

バス使用权の要求を説明する図、

【図18】

バス使用の許可を説明する図、

【図19】

1394シリアルバスにおけるアービトレーションの流れを示すフローチャート、

【図20】

トランザクションレイヤにおけるCSRアーキテクチャに基づくリード、ライト、ロックの各コマンドの要求・レスポンスプロトコルを示す図、

【図21】

リンクレイヤにおけるサービスを示す図、

【図22】

非同期転送における時間的な遷移を示す図、

【図23】

非同期転送用パケットのフォーマットを示す図、

【図24】

同期転送における時間的な遷移を示す図、

【図25】

同期転送用のパケットフォーマット例を示す図、

【図26】

1394シリアルバスにおける同期転送のパケットフォーマットのフィールドの詳細

細を示す図、

【図27】

同期転送と非同期転送が混在するときの転送状態の時間的遷移を示す図、

【図28A】

本実施形態のデジタルカメラにおける印刷処理を示すフローチャート、

【図28B】

本実施形態のプリンタにおける印刷処理を示すフローチャート、

【図29A】

第2実施形態のデジタルカメラにおける印刷処理を示すフローチャート、

【図29B】

第2実施形態のプリンタにおける印刷処理を示すフローチャート、

【図30A】

第3実施形態のデジタルカメラにおける印刷処理を示すフローチャート、

【図30B】

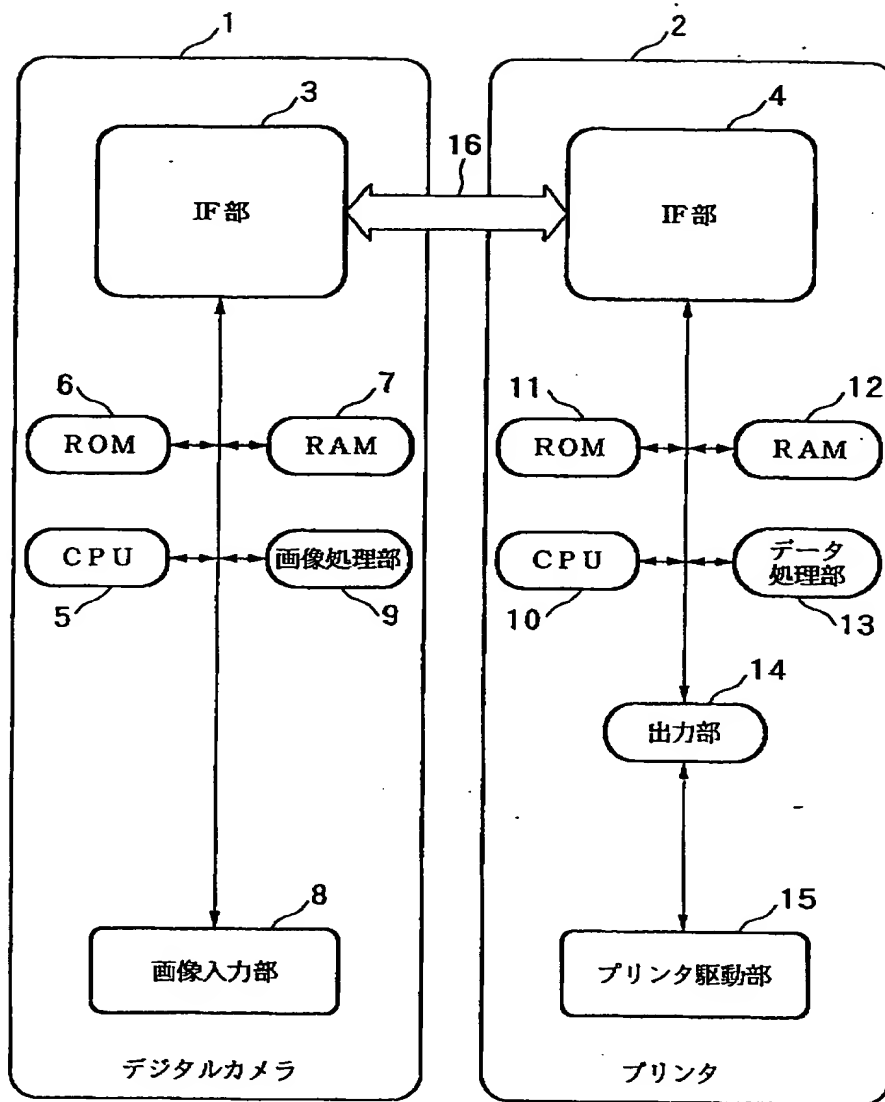
第3実施形態のプリンタにおける印刷処理を示すフローチャート、である。

【符号の説明】

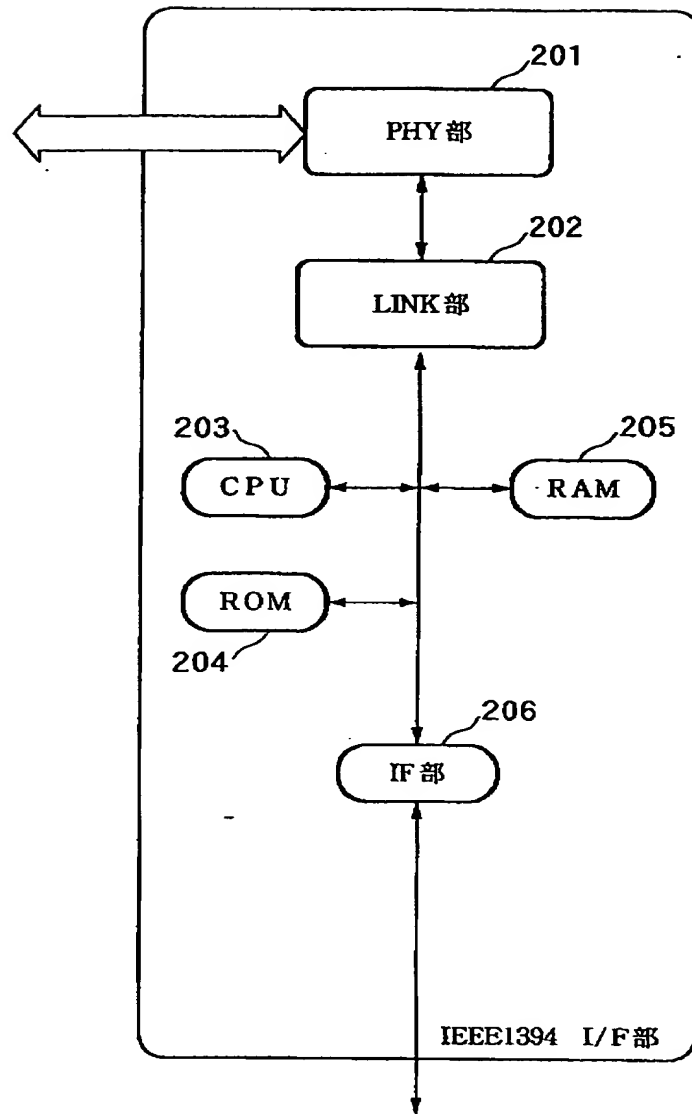
- 1 デジタルカメラ
- 2 プリンタ
- 3, 4 インタフェース部
- 5, 10 CPU
- 6, 11 ROM
- 7, 12 RAM
- 8 画像入力部
- 9 画像処理部
- 13 データ処理部
- 15 プリンタ駆動部
- 16 ケーブル

【書類名】 図面

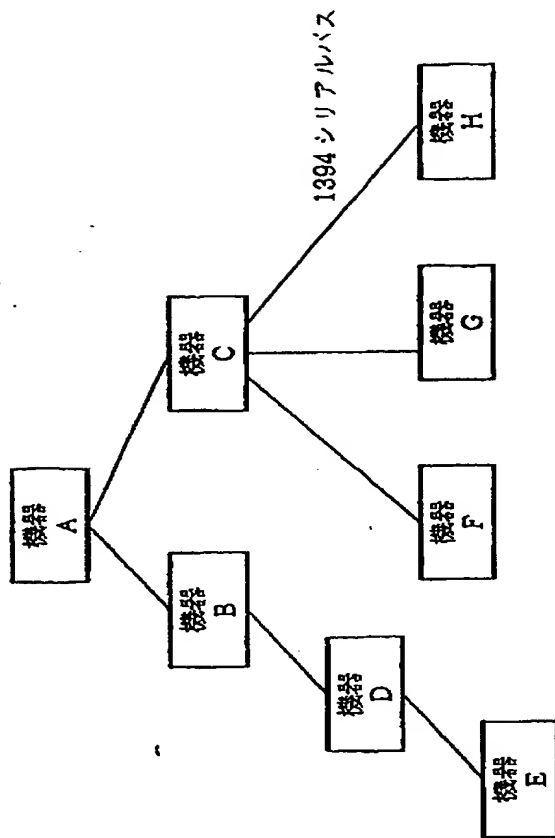
【図 1】



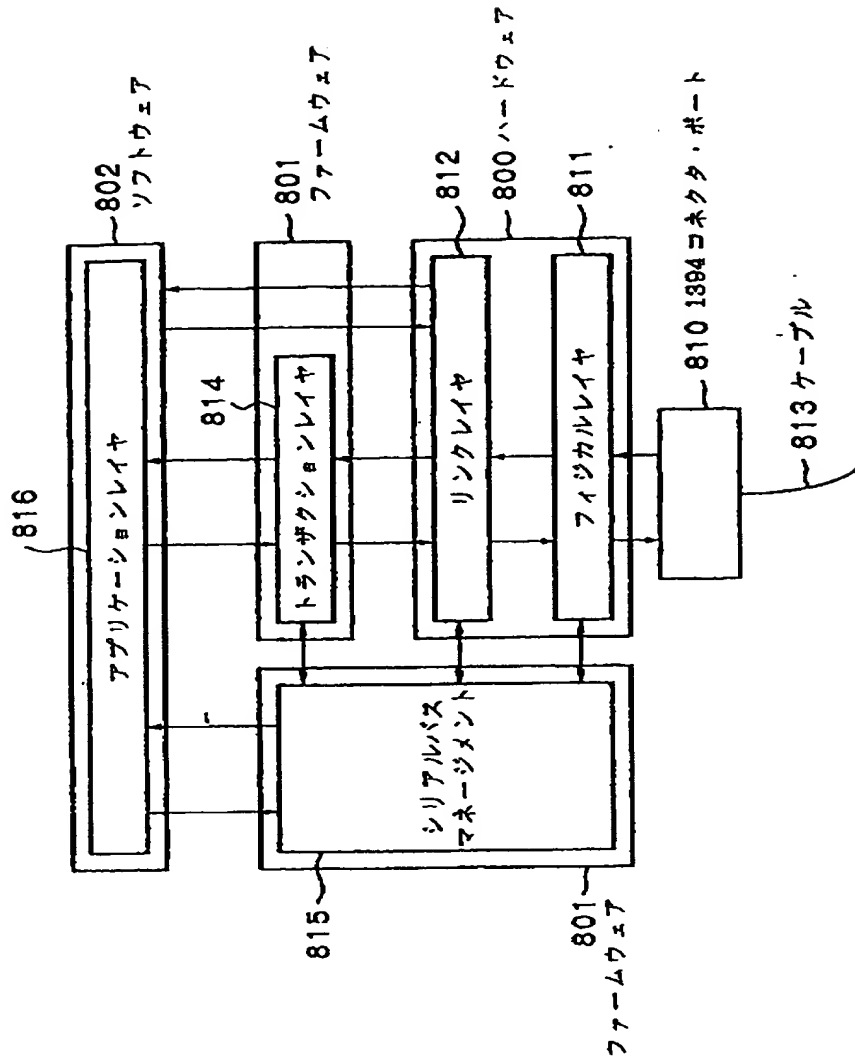
【図 2】



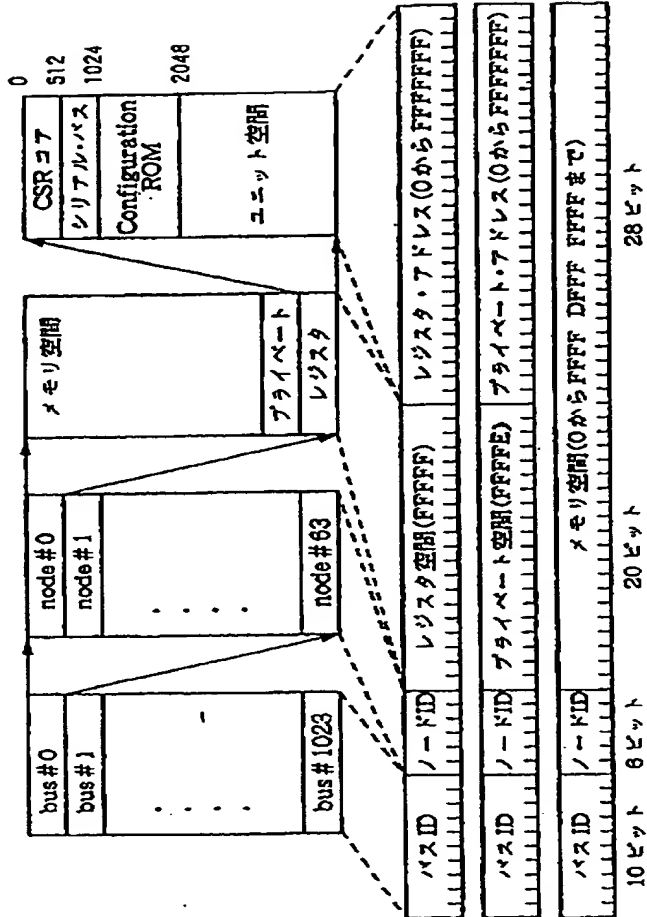
【図3】



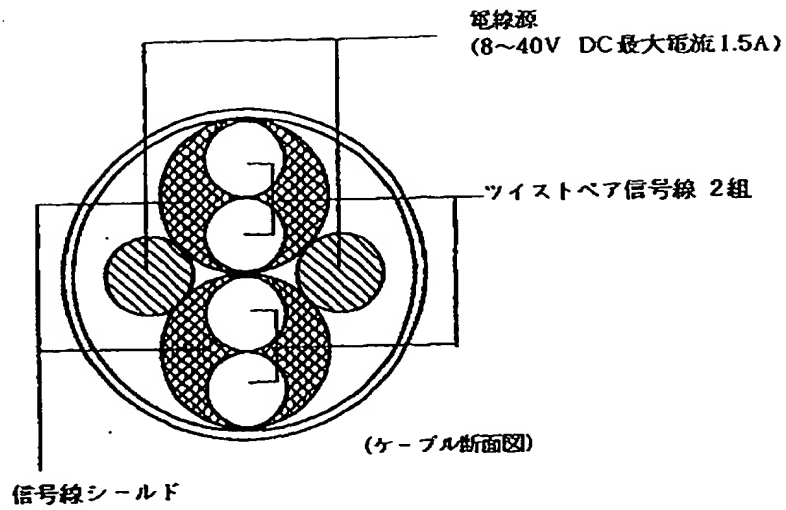
【図4】



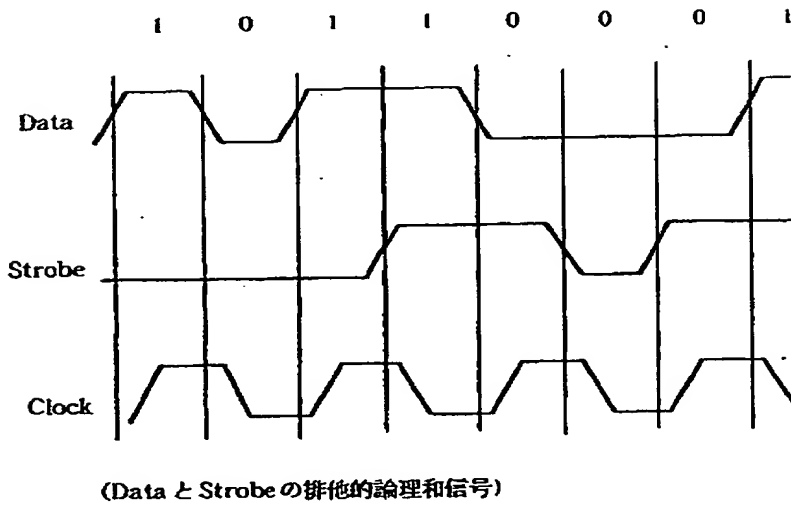
【図 5】



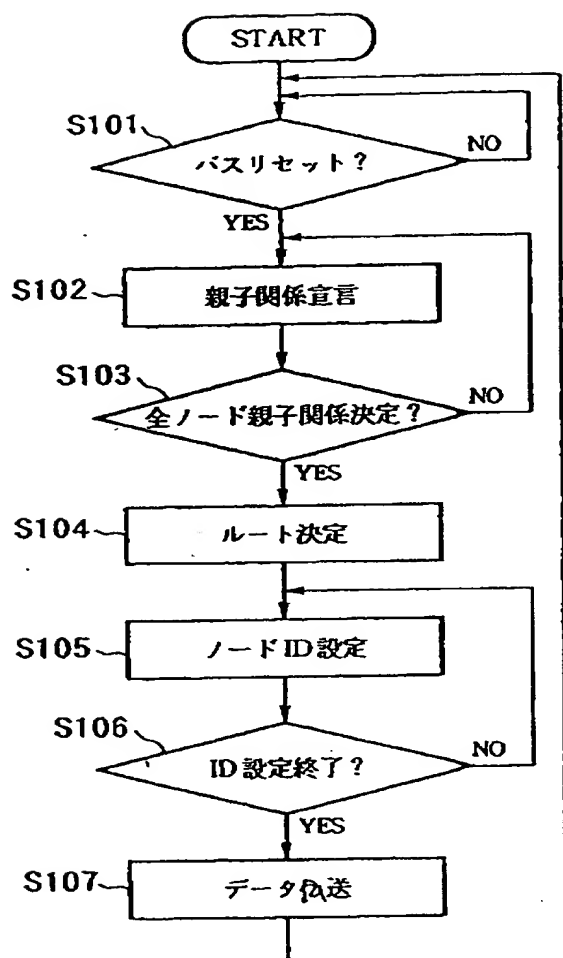
【図 6】



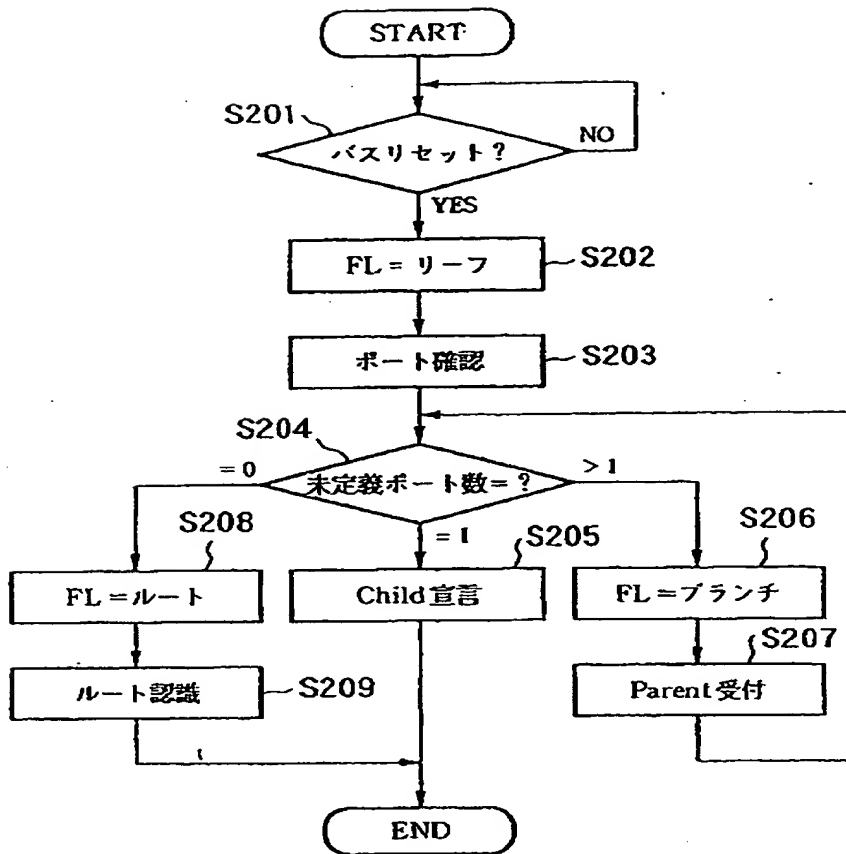
【図 7】



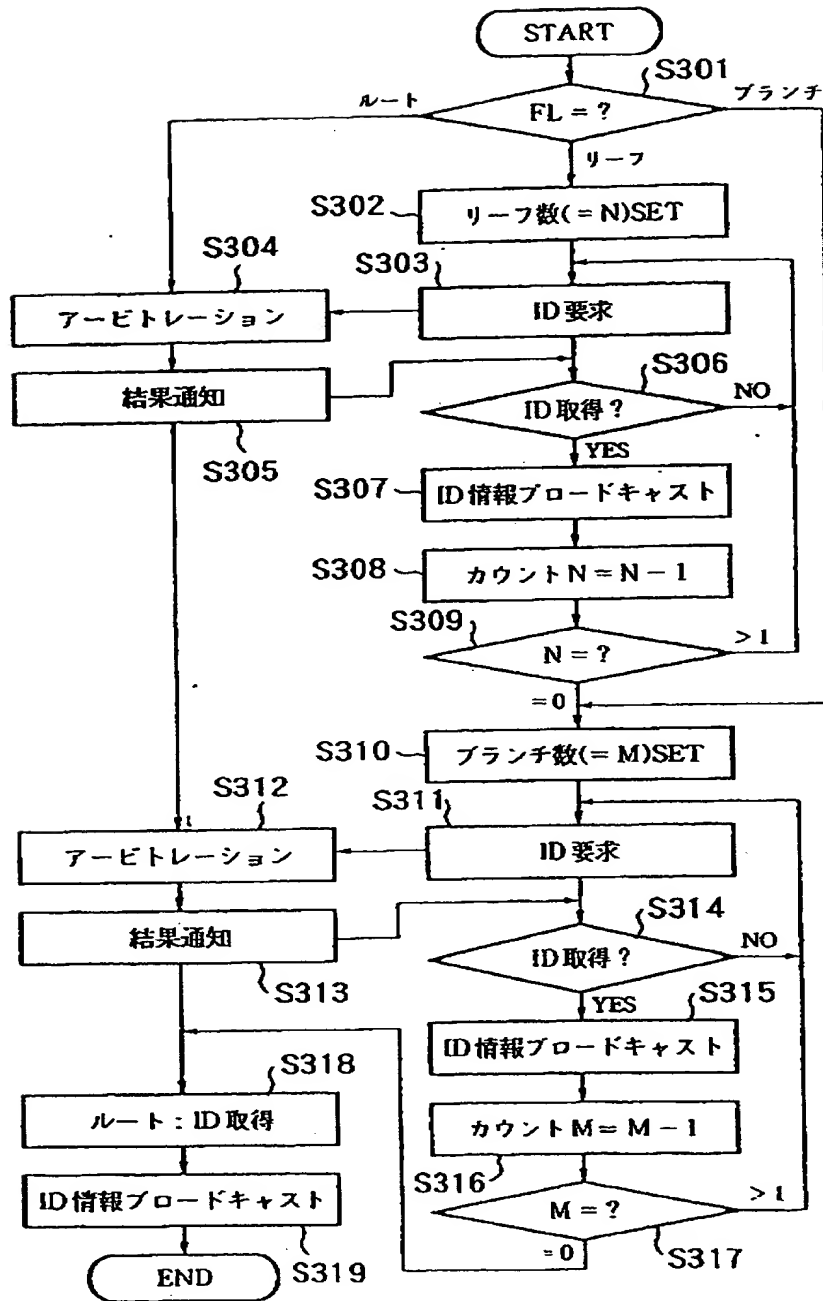
【図 8】



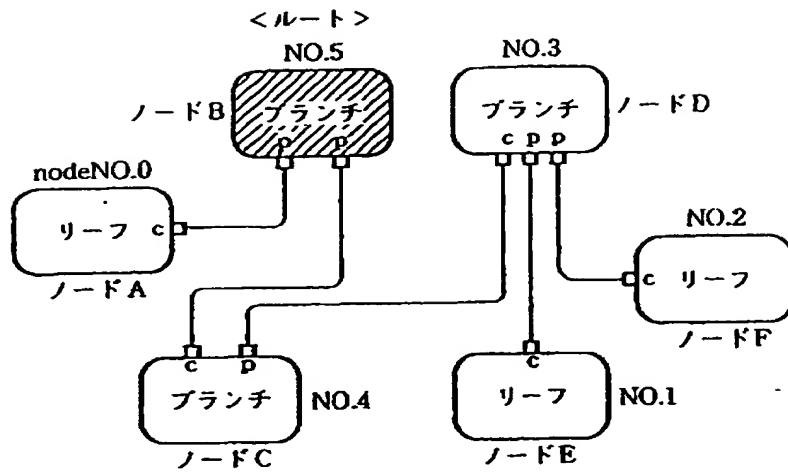
【図9】



【図10】



【図 1 1】



ブランチ：2つ以上のノード接続があるノード
リーフ：1つのポートのみ接続があるノード

□：ポート
c：子のノードに相当するポート
p：親のノードに相当するポート

【図12】

CSRコア・レジスタ

オフセット (16進数)	レジスタ名称	機 能
000	STATE_CLEAR	状態と制御の情報
004	STATE_SET	STATE_CLEARの書き込み可否を示す情報
008	NODE_IDS	バスID+ノードID
00C	RESET_START	この領域に対する書き込みでバスをリセット
010~014	INDIRECT_ADDRESS, INDIRECT_DATA	1Kより大きいROMをアクセスするための レジスタ
018~01C	SPLIT_TIMEOUT	スプリット・トランザクションのタイムアウトを 検出するタイマの値
020~02C	ARGUMENT, TEST_START, TEST_STATUS	診断用のレジスタ
030~04C	UNITS_BASE, UNITS_BOUND, MEMORY_BASE, MEMORY_BOUND	IEEE1394では、実施しない
050~054	INTERRUPT_TARGET, INTERRUPT_MASK	割り込み通知レジスタ
058~07C	CLOCK_VALUE, CLOCK_TICK_PERIOD, CLOCK_STORBE_ARRIVED, CLOCK_INFO	IEEE1394では、実施しない
080~0FC	MESSAGE_REQUEST, MESSAGE_RESPONSE	メッセージ通知レジスタ
100~17C		予約
180~1FC	ERROR_LOG_BUFFER	IEEE1394用に予約

【図 13】

シリアル・バス・レジスタ

オフセット (16進数)	レジスタ名称	機 能
200	CYCLE_TIME	アイソクロナス転送のためのカウンタ
204	BUS_TIME	時間を同期するためのレジスタ
208	POWER_FAIL_IMMINENT	電源供給に関するレジスタ
20C	POWER_SOURCE	
210	BUSY_TIMEOUT	トランザクション層の再試行を制御
214 }		予約
218		
21C	BUS_MANAGER_ID	バス・マネージャのノードID
220	BANDWIDTH_AVAILABLE	アイソクロナス転送の帯域を管理
224 }	CHANNELS_AVAILABLE	アイソクロナス転送のチャネル番号を管理
228		
22C	MAINT_CONTROL	診断用レジスタ
230	MAINT_UTILITY	
234 }		予約
3FC		

【図 1 4】

シリアル・バス装置レジスタ

オフセット (16進数)	レジスタ名称	機 能
800 ┆ FFC		予約
1000 ┆ 13FC	TOPOLOGY_MAP	シリアル・バスの構成情報
1400 ┆ 1FFC		予約
2000 ┆ 2FFC	SPEED_MAP	シリアル・バスの転送速度の 情報
3000 ┆ FFFC		予約

【図 1 5】

最小形式の Configuration ROM

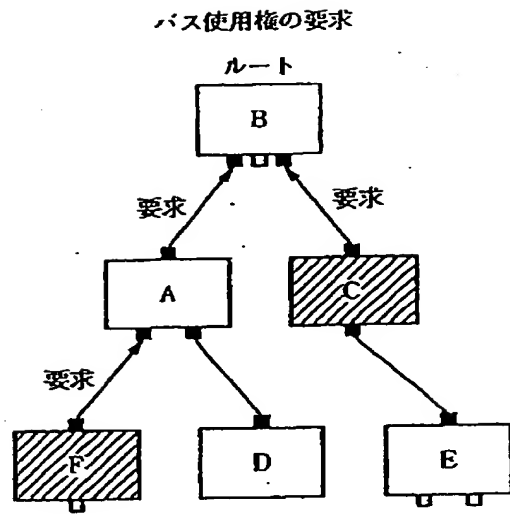
01	ベンダ ID
----	--------

【図 1 6】

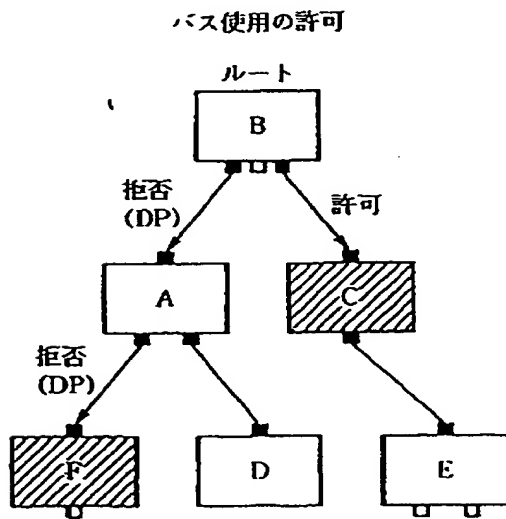
一般形式の Configuration ROM

bus_info_blockの長さ	ROMの長さ	CRC
bus_info_block (1394のASCIIコードと、ノードがアイソクロナス資源管理、 サイクル・マスダ、バス・マネージャの能力をもっているかの情報)		
root_directory (ベンダIDとノードの機能を表す)		
unit_directories (ユニットの種類とドライバ・ソフトのバージョン)		
root & unit leaves		
vender_dependent_information		

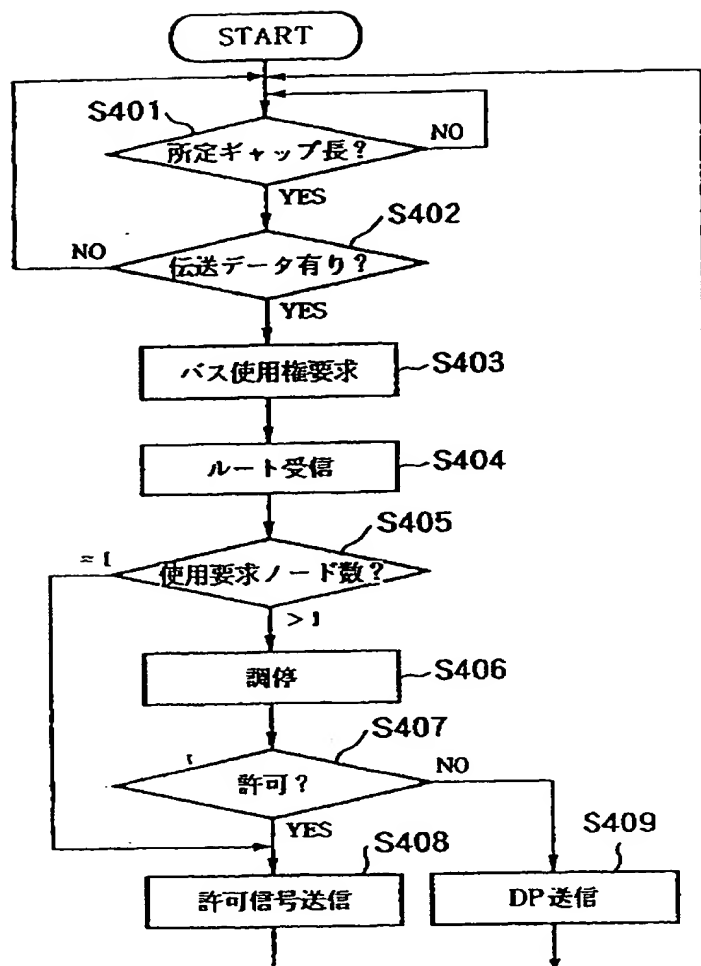
【図 17】



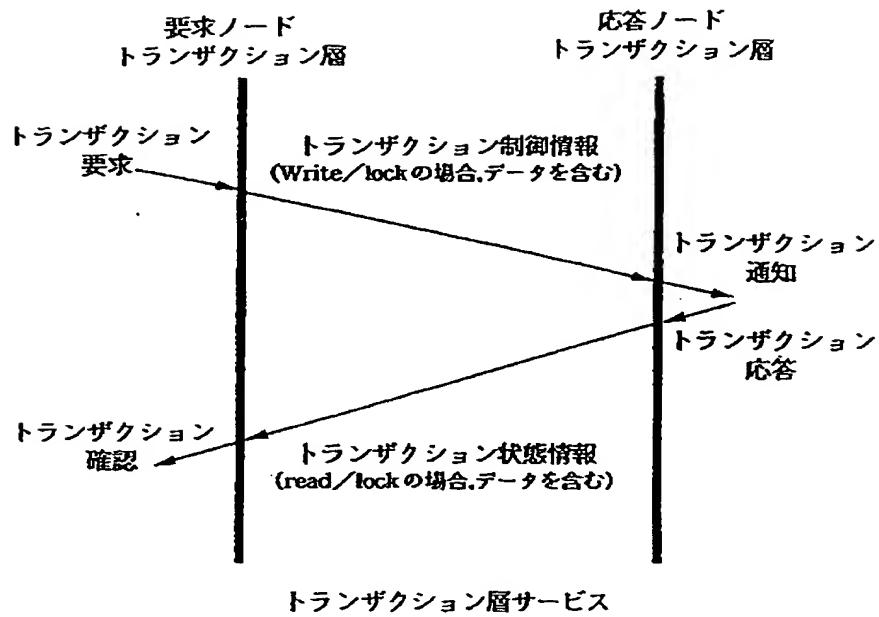
【図 18】



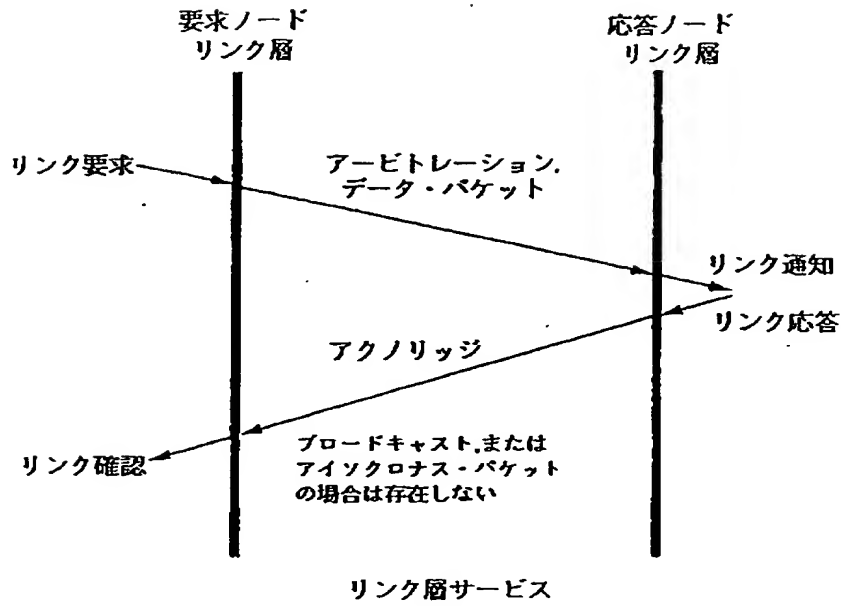
【図19】



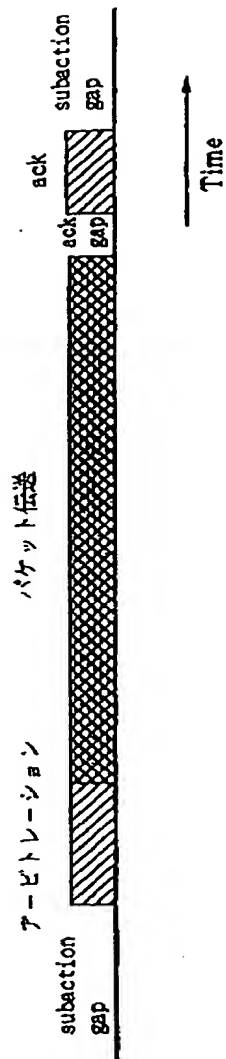
【図 20】



【図 21】



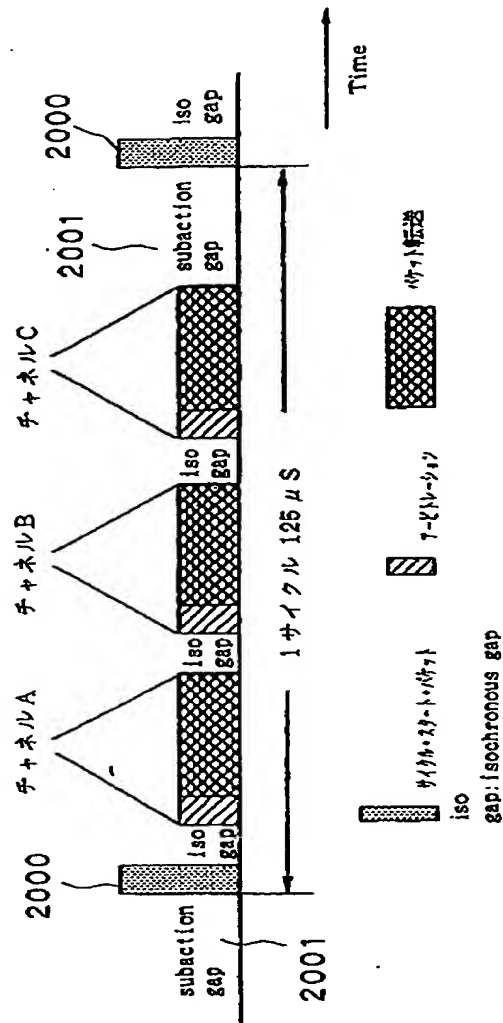
【図 22】



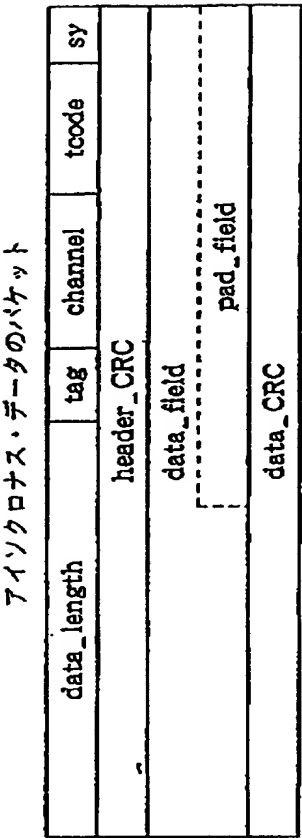
【図 23】

destination_ID	tl	rt	tcode	pri
source_ID	destination_offset			
data_length	extended_tcode			
header_CRC				
data_field				
pad_field				
data_CRC				

【図 24】



【図 25】

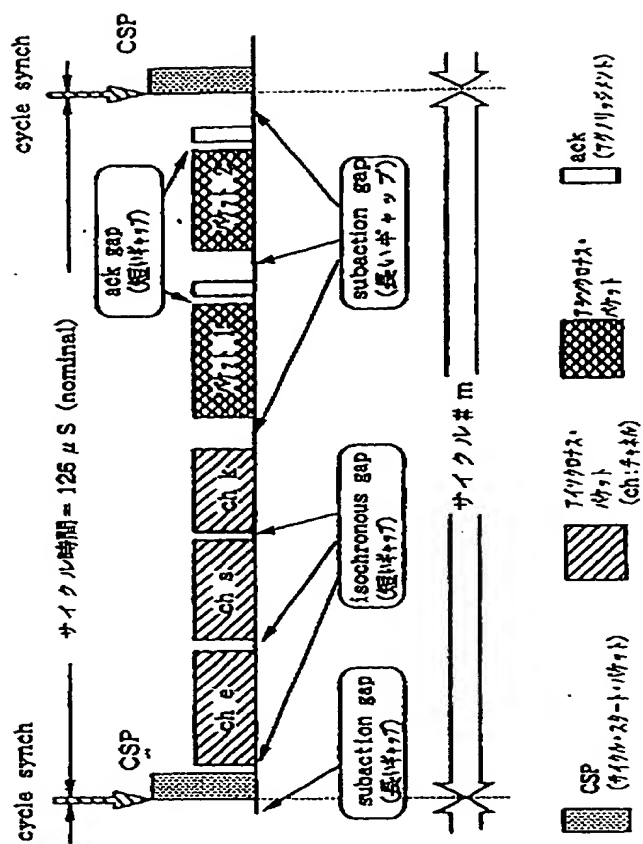


【図 26】

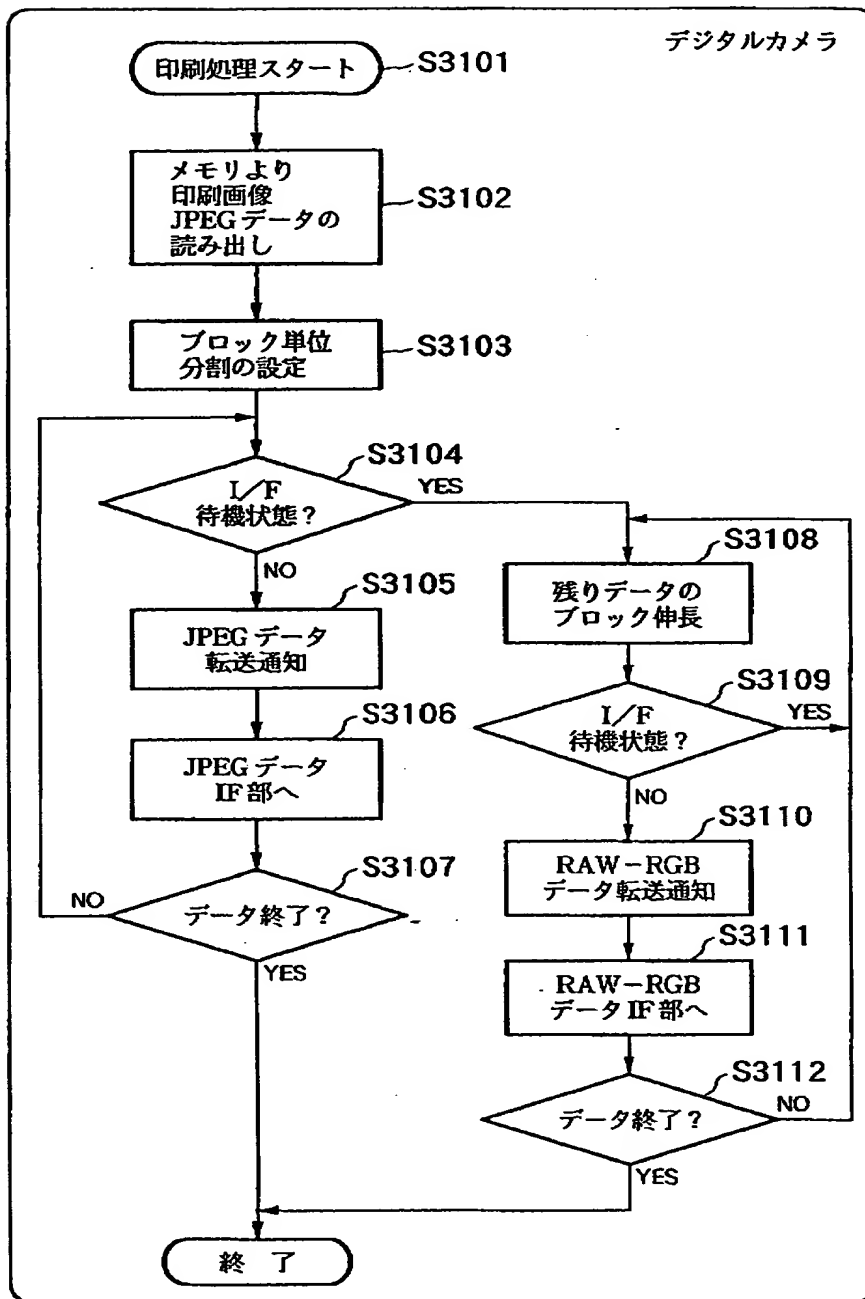
略称	名称	内容
destination_ID	destination identifier	着信先ノードのIDを示す (アシンクロナスのみ)
tl	transaction label	一連のトランザクションを示すためのラベル (アシンクロナスのみ)
rt	retry code	再送ステータスを示すコード (アシンクロナスのみ)
tcode	transcation code	パケットの種別を示すコード (アシンクロナスのみ)
pri	priority	優先順位 (アシンクロナスのみ)
source_ID	source identifier	発信元ノード (アシンクロナスのみ)
destination_offset	destination memory address	着信先ノードのメモリ・アドレス (アシンクロナスのみ)
rcode	response code	応答ステータス (アシンクロナスのみ)
quadiet_data	quadiet(4bytes) data	4バイト長のデータ (アシンクロナスのみ)
data_length	length of data	data_fieldの長さ (pad bytesは除く)
extended_tcode	extended transaction code	拡張トランザクション・コード (アシンクロナスのみ)
chanel	isochronous identifier	アイソクロナス・パケットの識別を行う
sy	synchronization code	映像・音声などの同期に使われる (アシンクロナスのみ)
cycle_time_data	contents of the CYCLE_TIME register	サイクル・マスタ・ノードのサイクル・タイマ・レジスタの値 (サイクル・パケットのみ)
data_field	data + pad bytes	データが格納される (アイソクロナスとアシンクロナス)
header_CRC	CRC for header field	ヘッダ部分に対するCRC
data_CRC	CRC for data field	データ部分に対するCRC
tag	tag label	アイソクロナス・パケットのフォーマット

o /

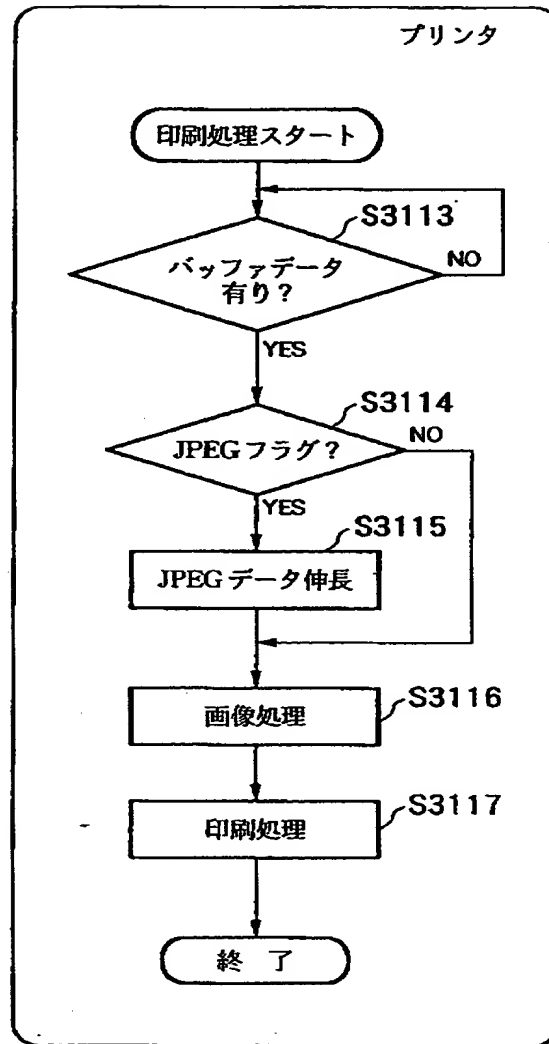
【図 27】



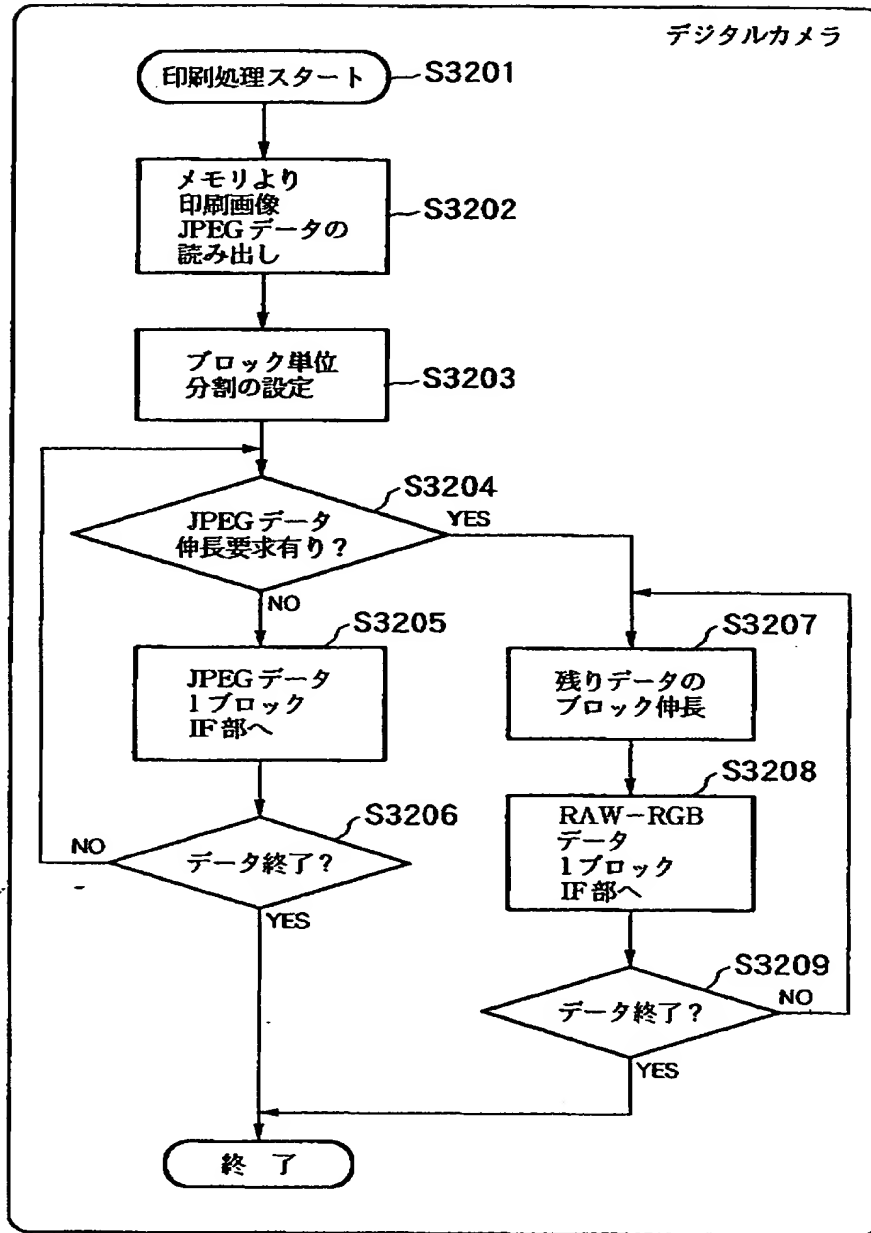
【図 28 A】



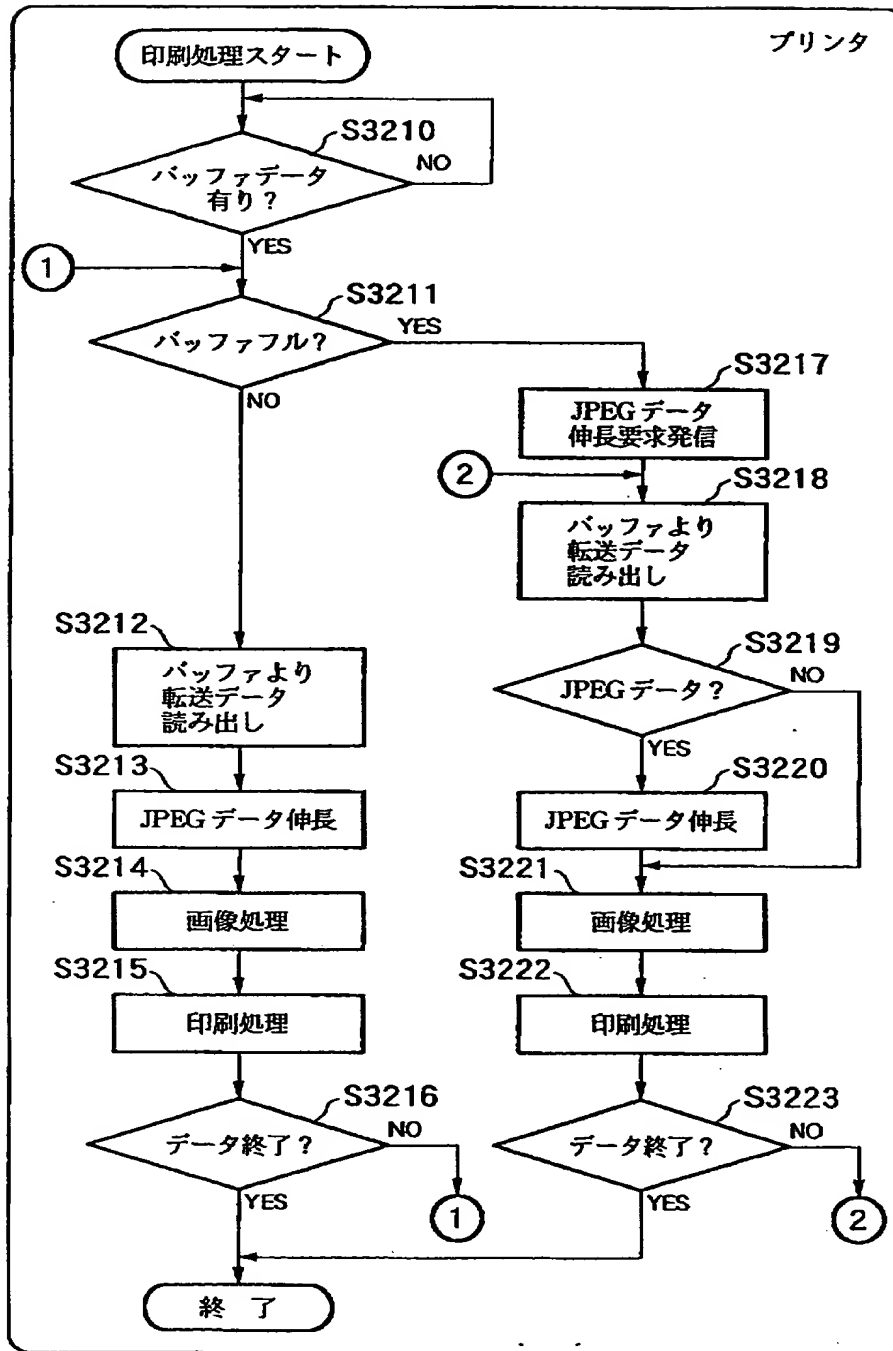
【図 2 8 B】



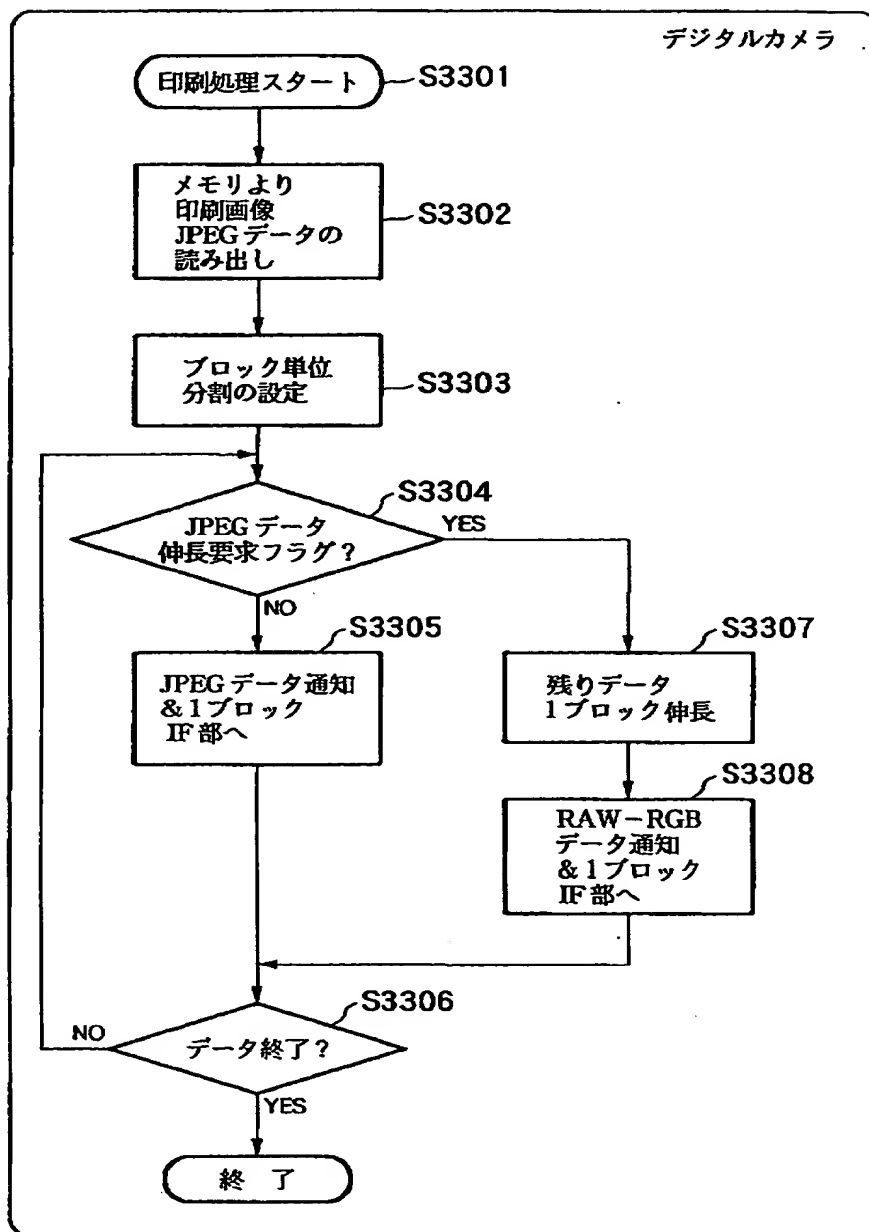
【図 29 A】



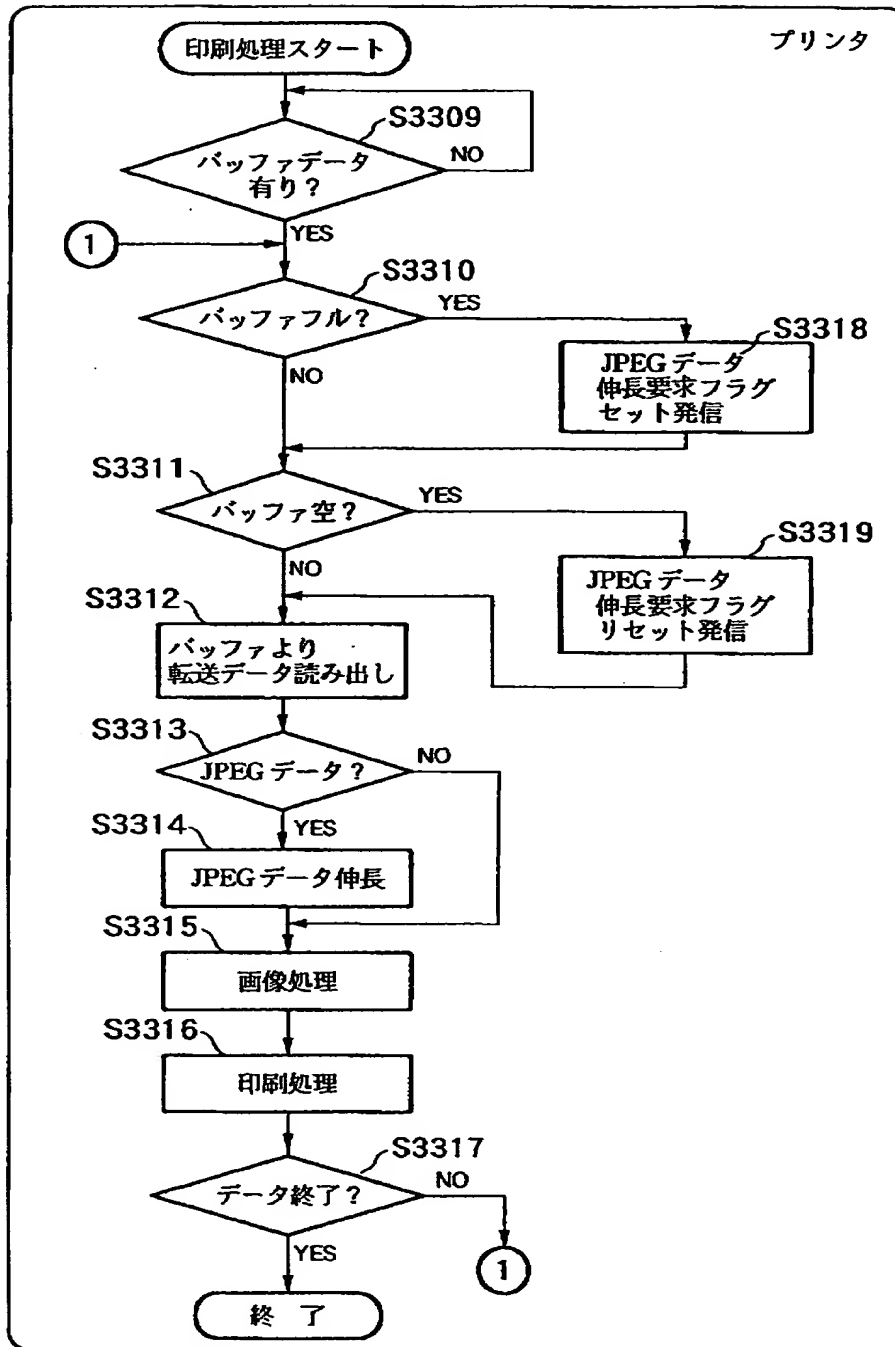
【図 29 B】



【図30A】



【図 30 B】



【書類名】 要約書

【要約】

【課題】 画像入力装置と画像形成装置とを接続したシステムにおいて、画像形成装置の動作中に画像入力装置は動作停止しており、機器間におけるパフォーマンスのバランスが悪いため、トータルスループットが向上しない。

【解決手段】 デジタルカメラにおいて印刷用のJPEGデータをプリンタに転送する際に、そのインタフェース状態を検出し(S3104)、転送が不可能なビジー状態となれば残りのJPEGデータを伸長してから(S3108)、転送を行う(S3110, S3111)

【選択図】 図 2 8 A

出 願 人 履 歴 情 報

識別番号 [000001007]

1. 変更年月日	1990年 8月30日
[変更理由]	新規登録
住 所	東京都大田区下丸子3丁目30番2号
氏 名	キヤノン株式会社

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.